Федеральное государственное бюджетное образовательное учреждение высшего образования
"Волжский государственный университет водного транспорта"

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ
Б.1.О.Д09 Информатика

Профиль	Эксплуатация судового электрооборудования и средств автоматики
Институт	Пермский филиал
	Отделение высшего образования
Специальность	26.05.07 Эксплуатация судового электрооборудования и средств автоматики

1. Объем дисциплины и виды учебной работы

Вид учебной работы	Всего часов/зачетных единиц	Курсы
		1
СРС	189 Часы	189
Лекции	6 Часы	6
Лабораторные работы	12 Часы	12
Экзамен	9 Часы	9

2. Место дисциплины в структуре ООП

Код дисциплины	Наименование блока	Трудоемкость дисциплины, з.е.
Б1.О.09	Блок 1 Дисциплины (модули) (Обязательная часть)	6

3. Разделы дисциплины и виды занятий

№ п/п	Наименование раздела дисциплины	СРС	Лекции	Лабораторные работы	Экзамен	Всего
1	Раздел 1. Современные информационные технологии и программные средства для решения задач профессиональной деятельности, обеспечение выполнения требований информационной безопасности (ОПК-5)	4 Часы				4 Часы
1,1	Тема 1.1. Информационно-телекоммуникационные технологии глобальной компьютерной сети Интернет. Структура и принципы функционирования глобальной сети Интернет.	4 Часы				4 Часы
1,2	Тема 1.2. Основы технологии искуственного интеллекта	10 Часы				10 Часы
1,3	Тема 1.3. Информационные технологии программирования на примере языка Си в среде MS Visual Studio 2010.	2 Часы				2 Часы
1,31	Тема 1.3.1. Разветвленные алгоритмы и их программирование.	2 Часы	1 Часы	1 Часы		4 Часы
1,31	Тема 1.3.1.1 Лабораторная работа "Разветвленные вычислительные процессы. Задача 1."	8 Часы				8 Часы
1,31	Тема 1.3.1.2. Лабораторная работа "Разветвленные вычислительные процессы. Задача 2."	2 Часы				2 Часы
1,31	Тема 1.3.1.3. Лабораторная работа "Разветвленные вычислительные процессы. Задача 3."	2 Часы				2 Часы
1,32	Тема 1.3.2. Циклические алгоритмы и их программирование.	2 Часы	0,5 Часы			2,5 Часы
1,32	Тема 1.3.2.1. Лабораторная работа "Исследование функции одной переменной"	4 Часы				4 Часы
1,32	Тема 1.3.2.2. Лабораторная работа "Обработка данных одномерного массива. Задача 1".	2 Часы	0,5 Часы	1 Часы		3,5 Часы
1,32	Тема 1.3.2.3. Лабораторная работа "Обработка данных одномерного массива. Задача 2."	2 Часы				2 Часы

1,32	Тема 1.3.2.4 Лабораторная работа "Обработка данных двумерного массива. Задача 1."	6 Часы				6 Часы
1,32	Тема 1.3.2.5 Лабораторная работа "Обработка данных двумерного массива. Задача 2."	6 Часы				6 Часы
1,4	Тема 1.4. Техническая эксплуатация информационных и телекоммуникационных систем на примере Microsoft office	5 Часы				5 Часы
1,41	Тема 1.4.1. Основные возможности текстового редактора MS Word.	15 Часы				15 Часы
1,42	Тема 1.4.2. Основные возможности табличного процессора MS Excel.	13 Часы	0,5 Часы	1 Часы		14,5 Часы
2	Раздел 2. Технологии обеспечения информационной безопасности. Методы и средства защиты информации. Антивирусная защита.	10 Часы				10 Часы
3	Раздел 3. Техническая эксплуатация информационных и телекоммуникационных систем на примере программирования микропроцессоров.	0,1 Часы				0,1 Часы
3,1	Тема 3.1. Структура МП, система команд МП. Команды пересылки. Арифметические команды.	0,1 Часы				0,1 Часы
3,2	Тема 3.2. Логические команды. Команды передачи управления.	0,1 Часы				0,1 Часы
3,3	Тема 3.3. Организация циклов. Алгоритм сложения. Адресация памяти.	0,1 Часы				0,1 Часы
3,31	Тема 3.3.1. Лабораторная работа "Сложение массива чисел". Часть 1.	0,1 Часы				0,1 Часы
3,32	Тема 3.3.2. Лабораторная работа "Сложение массива чисел". Часть 2.	0,1 Часы				0,1 Часы
3,33	Тема 3.3.3. Лабораторная работа "Использование приемов адресации".	0,1 Часы				0,1 Часы
3,4	Тема 3.4. Стек и работа подпрограмм.	0,1 Часы				0,1 Часы
3,41	Тема 3.4.1. Лабораторная работа "Вычисление выражения с использованием подпрограммы умножения однобайтных чисел ." Часть 1.	0,1 Часы				0,1 Часы
3,5	Тема 3.5. Индикация символов на дисплее МП.	0,1 Часы				0,1 Часы
3,6	Тема 3.6. Алгоритм бегущей строки.	0,1 Часы				0,1 Часы
3,61	Тема 3.6.1. Лабораторная работа "Организация движения текста на дисплее."	0,1 Часы				0,1 Часы
4	Раздел 4. Использование современных информационных технологий и программных средств при решении задач профессиональной деятельности на примере пакета MathCAD.	0,1 Часы				0,1 Часы
4,1	Тема 4.1. Решение типовых инженерных и математических задач.	0,1 Часы				0,1 Часы
4,11	Тема 4.1.1. Лабораторная работа "Основные инструменты MathCAD".		0,1 Часы	2 Часы		2,1 Часы
4,12	Тема 4.1.2. Лабораторная работа "Вычисление определенных интегралов".			1 Часы		1 Часы
4,13	Тема 4.1.3. Лабораторная работа "Обработка данных одномерных массивов".			1 Часы		1 Часы
4,14	Тема 4.1.4. Лабораторная работа "Сложные функции в MathCAD".		0,1 Часы	1 Часы		1,1 Часы
4,15	Тема 4.1.5. Лабораторная работа "Графики в MathCAD. Исследование функции одной переменной".		0,25 Часы	1 Часы		1,25 Часы
4,16	Тема 4.1.6. Лабораторная работа "Обработка данных двумерных массивов".	0,1 Часы				0,1 Часы
4,17	Тема 4.1.7. Лабораторная работа "Решение дифференциальных уравнений, описывающих переходные процессы в электрических цепях".	2 Часы	0,1 Часы	1 Часы		3,1 Часы
5	Раздел 5. Элементы компьютерного моделирования в информационных системах	0,1 Часы				0,1 Часы
5,1	Тема 5.1. Понятие моделирования, виды моделирования, модели, классификация, этапы математического моделирования	3,4 Часы				3,4 Часы
5,2	Тема 5.2. Моделирование процессов в электрических сетях. Неразветвленные сети.	8 Часы	0,2 Часы	1 Часы		9,2 Часы
5,3	Тема 5.3. Моделирование процессов в электрических сетях. Разветвленные сети.	8 Часы	0,25 Часы	1 Часы		9,25 Часы
5,4	Тема 5.4. Моделирование процессов в электрических сетях.Сети с нелинейными элементами.	4 Часы	0,25 Часы			4,25 Часы
5,5	Тема 5.5. Планирование, проведение и анализ результатов компьютерного эксперимента.	4 Часы				4 Часы
5,6	Тема 5.6. Моделирование различных типов входных сигналов в MathCAD.	4 Часы				4 Часы

6	Раздел 6. Безопасное техническое использование компьютерной информационной системы в соответствии с международными и национальными требованиями	10 Часы	2 Часы			12 Часы
6,1	Тема 6.1. Современные судовые компьютерные нформационные системы, их назначение, основные модули.	6 Часы				6 Часы
6,2	Тема 6.2. Основные международные и национальные требования по безопасной эксплуатации компьютерной информационной системы.	10 Часы	0,25 Часы			10,25 Часы
6,3	Тема 6.3. Информационная безопасность в глобальных сетях.	8 Часы				8 Часы
6,4	Тема 6.4. Анализ способов нарушений информационной безопасности	6 Часы				6 Часы
6,5	Тема 6.5. Общие положения ТЗИ. Физические средства защиты.	6 Часы				6 Часы
6,6	Тема 6.6. Защита информации от несанкционированного доступа.	9 Часы				9 Часы

4. Компетенции обучающегося, формируемые в процессе освоения дисциплины

№ п/п	Наименование раздела дисциплины	Формируемые компетенции	
1	Раздел 1. Современные информационные технологии и программные средства для решения задач профессиональной деятельности, обеспечение выполнения требований информационной безопасности (ОПК-5)	ОПК-5(ИДК: ОПК-5.3.1;)	
1,1	Тема 1.1. Информационно-телекоммуникационные технологии глобальной компьютерной сети Интернет. Структура и принципы функционирования глобальной сети Интернет.		
1,2	Тема 1.2. Основы технологии искуственного интеллекта		
1,3	Тема 1.3. Информационные технологии программирования на примере языка Си в среде MS Visual Studio 2010.		
1,31	Тема 1.3.1. Разветвленные алгоритмы и их программирование.		
1,31	Тема 1.3.1.1 Лабораторная работа "Разветвленные вычислительные процессы. Задача 1."		
1,31	Тема 1.3.1.2. Лабораторная работа "Разветвленные вычислительные процессы. Задача 2."		
1,31	Тема 1.3.1.3. Лабораторная работа "Разветвленные вычислительные процессы. Задача 3."		
1,32	Тема 1.3.2. Циклические алгоритмы и их программирование.		
1,32	Тема 1.3.2.1. Лабораторная работа "Исследование функции одной переменной"		
1,32	Тема 1.3.2.2. Лабораторная работа "Обработка данных одномерного массива. Задача 1".		
1,32	Тема 1.3.2.3. Лабораторная работа "Обработка данных одномерного массива. Задача 2."		
1,32	Тема 1.3.2.4 Лабораторная работа "Обработка данных двумерного массива. Задача 1."		
1,32	Тема 1.3.2.5 Лабораторная работа "Обработка данных двумерного массива. Задача 2."		
1,4	Тема 1.4. Техническая эксплуатация информационных и телекоммуникационных систем на примере Microsoft office	ОПК-5(ИДК: ОПК-5.3.1;)	
1,41	Тема 1.4.1. Основные возможности текстового редактора MS Word.		
1,42	Тема 1.4.2. Основные возможности табличного процессора MS Excel.		
2	Раздел 2. Технологии обеспечения информационной безопасности. Методы и средства защиты информации. Антивирусная защита.		
3	Раздел 3. Техническая эксплуатация информационных и телекоммуникационных систем на примере программирования микропроцессоров.	ОПК-5(ИДК: ОПК-5.3.1;)	

3,1	Тема 3.1. Структура МП, система команд МП. Команды пересылки. Арифметические команды.	ОПК-5(ИДК: ОПК-5.3.1;)	
3,2	Тема 3.2. Логические команды. Команды передачи управления.	ОПК-5(ИДК: ОПК-5.3.1;)	
3,3	Тема 3.3. Организация циклов. Алгоритм сложения. Адресация памяти.	ОПК-5(ИДК: ОПК-5.3.1;)	
3,31	Тема 3.3.1. Лабораторная работа "Сложение массива чисел". Часть 1.	ОПК-5(ИДК: ОПК-5.3.1;)	
3,32	Тема 3.3.2. Лабораторная работа "Сложение массива чисел". Часть 2.	ОПК-5(ИДК: ОПК-5.3.1;)	
3,33	Тема 3.3.3. Лабораторная работа "Использование приемов адресации".	ОПК-5(ИДК: ОПК-5.3.1;)	
3,4	Тема 3.4. Стек и работа подпрограмм.	ОПК-5(ИДК: ОПК-5.3.1;)	
3,41	Тема 3.4.1. Лабораторная работа "Вычисление выражения с использованием подпрограммы умножения однобайтных чисел ." Часть 1.	ОПК-5(ИДК: ОПК-5.3.1;)	
3,5	Тема 3.5. Индикация символов на дисплее МП.	ОПК-5(ИДК: ОПК-5.3.1;)	
3,6	Тема 3.6. Алгоритм бегущей строки.	ОПК-5(ИДК: ОПК-5.3.1;)	
3,61	Тема 3.6.1. Лабораторная работа "Организация движения текста на дисплее."	ОПК-5(ИДК: ОПК-5.3.1;)	
4	Раздел 4. Использование современных информационных технологий и программных средств при решении задач профессиональной деятельности на примере пакета MathCAD.	ОПК-5(ИДК: ОПК-5.3.1;)	
4,1	Тема 4.1. Решение типовых инженерных и математических задач.	ОПК-5(ИДК: ОПК-5.3.1;)	
4,11	Тема 4.1.1. Лабораторная работа "Основные инструменты MathCAD".		
4,12	Тема 4.1.2. Лабораторная работа "Вычисление определенных интегралов".		
4,13	Тема 4.1.3. Лабораторная работа "Обработка данных одномерных массивов".		
4,14	Тема 4.1.4. Лабораторная работа "Сложные функции в MathCAD".		
4,15	Тема 4.1.5. Лабораторная работа "Графики в MathCAD. Исследование функции одной переменной".		
4,16	Тема 4.1.6. Лабораторная работа "Обработка данных двумерных массивов".	ОПК-5(ИДК: ОПК-5.3.1;)	
4,17	Тема 4.1.7. Лабораторная работа "Решение дифференциальных уравнений, описывающих переходные процессы в электрических цепях".		
5	Раздел 5. Элементы компьютерного моделирования в информационных системах	ОПК-5(ИДК: ОПК-5.3.1;)	
5,1	Тема 5.1. Понятие моделирования, виды моделирования, модели, классификация, этапы математического моделирования		
5,2	Тема 5.2. Моделирование процессов в электрических сетях. Неразветвленные сети.		
5,3	Тема 5.3. Моделирование процессов в электрических сетях. Разветвленные сети.		
5,4	Тема 5.4. Моделирование процессов в электрических сетях.Сети с нелинейными элементами.		
5,5	Тема 5.5. Планирование, проведение и анализ результатов компьютерного эксперимента.		
5,6	Тема 5.6. Моделирование различных типов входных сигналов в MathCAD.		
6	Раздел 6. Безопасное техническое использование компьютерной информационной системы в соответствии с международными и национальными требованиями	ПК-6.(ИДК: ПК-6.3.1;)	
6,1	Тема 6.1. Современные судовые компьютерные нформационные системы, их назначение, основные модули.	ПК-6.(ИДК: ПК-6.3.1;)	
6,2	Тема 6.2. Основные международные и национальные требования по безопасной эксплуатации компьютерной информационной системы.	ПК-6.(ИДК: ПК-6.3.1;)	
6,3	Тема 6.3. Информационная безопасность в глобальных сетях.	ПК-6.(ИДК: ПК-6.3.1;)	
6,4	Тема 6.4. Анализ способов нарушений информационной безопасности	ПК-6.(ИДК: ПК-6.3.1;)	
6,5	Тема 6.5. Общие положения ТЗИ. Физические средства защиты.	ПК-6.(ИДК: ПК-6.3.1;)	
6,6	Тема 6.6. Защита информации от несанкционированного доступа.	ПК-6.(ИДК: ПК-6.3.1;)	

5. Перечень планируемых результатов обучения, соотнесенных с планируемыми результатами освоения

Процесс изучения дисциплины направлен на формирование и развитие у обучающегося следующих компетенций:

№ п/п	компетенции	Индикатор достижения		
		Знать	Уметь	Владеть
1	Способен использовать современные информационные технологии и программные средства при решении задач профессиональной деятельности, обеспечивая выполнение требований информационной безопасности	ОПК-53.1 принципы работы современных информационных технологий	ОПК-5У.1 применять современные информационные технологии и использовать их для решения задач профессиональной деятельности	ОПК-5В.1 навыками применения современных информационных технологий и использования их для решения задач профессиональной деятельности
2	Способен осуществлять безопасное техническое использование, техническое обслуживание судовой компьютерной информационной системы в соответствии с международными и национальными требованиями	ПК-6.3.1 международные и национальные требования, учитываемые при эксплуатации судовой компьютерной информационной системы	ПК-6.У.1 осуществлять безопасное техническое использование, техническое обслуживание судовой компьютерной информационной системы	ПК-6.В.1 навыками безопасного технического использования, технического обслуживания судовой компьютерной информационной системы

Процесс изучения дисциплины направлен на формирование и развитие у обучающегося следующих требуемых Международной конвенцией о подготовке и дипломировании моряков и несении вахты 1978 года стандартов компетентности:

№ п/п	Таблица	Функция	Сфера компетентности
1		A-III/6-1. Электрооборудование, электронная аппаратура и системы управления на уровне эксплуатации	A-III/6-1.5. Эксплуатация компьютеров и компьютерных сетей на судах

6. Материально-техническое и учебно-методическое обеспечение программы

6.1 Помещения и оборудование

№ п/п	Вид помещений	Оснащение помещений	№ помещений
1	Учебные аудитории для проведения учебных занятий	оборудование и технические средства обучения (Стул (25 ед.); Стол компьютерный (15 ед.); Системный блок (1 ед.); Монитор (1 ед.); Ноутбук (13 ед.); Проектор (1 ед.); Коммутатор (1 ед.); Доска (1 ед.); Экран (1 ед.) (463) Парты (37 ед.); Мультимедийное оборудование (1 ед.) (570))	463,562
2	Помещения для самостоятельной работы обучающихся	компьютерная техника с возможностью подключения к сети "Интернет" и обеспечение доступа в электронную информационно-образовательную среду университета	463

6.2 Лицензионное и свободно распространяемое программное обеспечение, в том числе отечественного производства

№ п/п	Наименование
1	Microsoft Office Professional Plus 2016 (Договор №44/109-15 от 28.12.2015 (бессрочно))
2	Microsoft Office ProPlus 2013 (Договор №44/59-18 от 09.04.2018 (бессрочно))

6.3 Карта обеспеченности печатными и(или) электронными изданиями и электронными образовательными ресурсами

№ п/п	Наименование источника	Год издания	Ресурс	Количество экземпляров
1	Крайнова, В.В. Методические указания по организации и выполнению внеаудиторной (самостоятельной) работы [Электронный ресурс] : для преподавателей и студ.по направлениям подготовки (спец.) высш.и сред.проф.образования / В. В. Крайнова ; ВГУВТ. - Н.Новгород, 2018. - 1 текст/файл. - Авторский вариант. – Режим доступа: http://94.100.87.24:8080/MarcWeb/Tmp/f15520.pdf	2 018	Электронный ресурс	0
2	Кудинов, Ю.И.;Основы современной информатики;учеб.пособие;Кудинов, Ю.И.Пащенко, Ф.Ф.-Санкт-Петербург,Лань; URL: https://reader.lanbook.com/book/213647#1 (дата обращения: 24.05.2022) ;<null>	2 022	Электронный ресурс	0
3	Сергеев, А.Н.;Основы локальных компьютерных сетей;учебное пособие для вузов;Сергеев, А.Н.-СПб.,Лань; URL: https://e.lanbook.com/book/152651 (дата обращения: 23.09.2021) ;<null>	2 021	Электронный ресурс	0

4	Гаврилова, Т.И.;Информатика.MATHCAD;учебно-метод.пособие для студ.очн.и заочн.обучения всех спец.и направлений подготовки;Гаврилова, Т.И.Шеянов, А.В.-Н.Новгород,<null>; <null> ;http://94.100.87.24:8080/marcweb/	2 019	Электронный ресурс	0
5	Мойзес, О.Е.;Информатика.Углубленный курс;учебное пособие для вузов;Кузьменко, Е.А.Мойзес, О.Е.-Москва,Юрайт; URL: https://urait.ru/viewer/informatika-uglublennyy-kurs-490342#page/1 (дата обращения 11.09.2022) ;<null>	2 022	Электронный ресурс	0
6	Тузовский, А.Ф.;Объектно-ориентированное программирование;учебное пособие для вузов;Тузовский, А.Ф.-Москва,Юрайт; URL: https://urait.ru/bcode/537332 (дата обращения: 16.04.2024) ;<null>	2 024	Электронный ресурс	0
7	Гаврилов, М.В.;Информатика и информационные технологии;учебник для вузов;Гаврилов, М.В.Климов, В.А.-Москва,Юрайт; URL: https://urait.ru/viewer/informatika-i-informacionnye-tehnologii-468473#page/1 (дата обращения: 27.09.2021). - Режим доступа: для авторизированных пользователей ;<null>	2 021	Электронный ресурс	0
8	Трофимов, В.В.;Информатика;учебник для вузов:В 2 томах;Барабанова, М.И.Ильина, О.П.Кияев, В.И.Минаков, В.Ф.Павловская, Т.А.Пушкина, Н.В.Саитов, А.В.Трофимов, В.В.-Москва,Юрайт; URL: https://urait.ru/viewer/informatika-v-2-t-tom-2-470745#page/1 (дата обращения: 22.11.2021) ;<null>	2 021	Электронный ресурс	0

Программа предусматривает возможность применения электронного обучения, дистанционных образовательных технологий.

Электронная информационно-образовательная среда университета с возможностью доступа к информационно- телекоммуникационной сети "Интернет" - Режим доступа: http://www.eios.vsuwt.ru/.

6.4 Современные профессиональные базы данных

№ п/п	Наименование
1	Статистический сборник: Транспорт в России- Режим доступа: http://www.gks.ru/wps/wcm/connect/rosstat_main/rosstat/ru/statistics/publications/catalog/doc_1136983505312
2	Центральная база статистических данных - Режим доступа: http://cbsd.gks.ru/

6.5 Информационные справочные системы

№ п/п	Наименование
1	Справочная правовая система «КонсультантПлюс» - Режим доступа: http://www.consultant.ru (договор от 02.02.2015 г.)
2	Справочная правовая система «Гарант» - Режим доступа: http://www.garant.ru (договор 62/16 от 01.09.2016 г. - бессрочный)
3	Справочная правовая система «КонсультантПлюс» - Режим доступа: http://www.consultant.ru (договор от 02.02.2015 г.)
4	Справочная правовая система «Гарант» - Режим доступа: http://www.garant.ru (договор 62/16 от 01.09.2016 г. - бессрочный)

7. Оценочные и методические материалы

Оценочные и методические материалы, определяющие процедуры оценивания индикаторов, характеризующих этапы формирования компетенций, являются приложением к программе.

№ п/п	Индикатор достижения компетенций	Сфера компетентности (МК ПДНВ)	Контролируемые разделы (темы)	Формы и методы контроля и оценки результатов обучения		Процедура оценивания	Критерии оценивания результата обучения и шкала оценивания			
							2	3	4	5
				Вид контроля	Форма контроля		не зачтено	зачтено		

1	ОПК-5З.1, ОПК-5У.1, ОПК-5В.1, ПК-6.З.1,ПК-6.У.1,ПК-6.В.1	А-III/6-1.5.	1.1.1, 1.1.2, 1.1.3, 1.2, 1.2.1, 1.2.1.1, 1.2.1.2, 1.2.1.3, 1.2.2, 1.2.2.1, 1.2.2.2, 1.2.2.3, 1.2.2.4, [illegible] 5	Текущий контроль	Лабораторная работа	Собеседование по лабораторным работам по дисциплине "Информатика", По результатам выполнения работ создается отчет. Образец отчета прилагается.	Работа выполнена не полностью и объем выполненной части работы не позволяет сделать правильных выводов: если опыты, измерения, вычисления, наблюдения производились неправильно	Работа выполнена не полностью, но объем выполненной части позволяет получить правильные результаты и выводы, если в ходе проведения опыта, измерений, вычислений и наблюдений были допущены ошибки	Работа выполнена в полном объеме с соблюдением необходимой последовательности проведения опытов, измерений, вычислений и наблюдений; все опыты проводит в условиях и режимах, обеспечивающих получение правильных результатов и выводов; в отчете правильно и аккуратно выполняет все записи, таблицы, рисунки, чертежи, графики, вычисления; но допускает несколько недочетов, оформление отчета согласно установленным требованиям; при защите отчета обучающийся ответил в основном на все контрольные вопросы	Работа выполнена в полном объеме с соблюдением необходимой последовательности проведения опытов, измерений, вычислений и наблюдений; все опыты проводит в условиях и режимах, обеспечивающих получение правильных результатов и выводов; в отчете правильно и аккуратно выполняет все записи, таблицы, рисунки, чертежи, графики, вычисления; оформление отчета согласно установленным требованиям; при защите отчета обучающийся ответил на все контрольные вопросы

2	ОПК-5З.1, ОПК-5У.1, ОПК-5В.1, ПК-6.З.1,ПК-6.У.1,ПК-6.В.1	А-III/6-1.5.	[illegible] 5	Текущий контроль	Лабораторная работа	Защита отчета по лабораторной работе.	Работа выполнена не полностью и объем выполненной части работы не позволяет сделать правильных выводов: если опыты, измерения, вычисления, наблюдения производились неправильно	Работа выполнена не полностью, но объем выполненной части позволяет получить правильные результаты и выводы, если в ходе проведения опыта, измерений, вычислений и наблюдений были допущены ошибки	Работа выполнена в полном объеме с соблюдением необходимой последовательности проведения опытов, измерений, вычислений и наблюдений; все опыты проводит в условиях и режимах, обеспечивающих получение правильных результатов и выводов; в отчете правильно и аккуратно выполняет все записи, таблицы, рисунки, чертежи, графики, вычисления; но допускает несколько недочетов, оформление отчета согласно установленным требованиям; при защите отчета обучающийся ответил в основном на все контрольные вопросы	Работа выполнена в полном объеме с соблюдением необходимой последовательности проведения опытов, измерений, вычислений и наблюдений; все опыты проводит в условиях и режимах, обеспечивающих получение правильных результатов и выводов; в отчете правильно и аккуратно выполняет все записи, таблицы, рисунки, чертежи, графики, вычисления; оформление отчета согласно установленным требованиям; при защите отчета обучающийся ответил на все контрольные вопросы

3	ОПК-5З.1, ОПК-5У.1, ОПК-5В.1, ПК-6.З.1,ПК-6.У.1,ПК-6.В.1	А-III/6-1.5.	5	Промежуточная аттестация	Экзамен	Экзамен за 1 семестр. 60 минут на подготовку к экзамену. Билет содержит два вопроса из предложенного перечня.	Незнание или непонимание обучающимся основного материала; на большую часть вопросов по содержанию экзамена затрудняется дать ответ или не дает верных ответов	Знания имеют фрагментарный характер, отличаются поверхностностью и малой содержательностью; содержание билета раскрыто слабо, имеются неточности при ответе на основные вопросы билета; нарушена логика изложения, отсутствует осмысленность представляемого материала	Знания имеют достаточный содержательный уровень, однако отличаются слабой структурированностью; раскрыто содержание билета, имеются неточности при ответе на дополнительные вопросы; недостаточно раскрыта проблема по одному из вопросов билета	Знания отличаются глубиной и содержательностью, дается полный исчерпывающий ответ, как на основные вопросы билета, так и на дополнительные; обучающийся свободно владеет научными понятиями; логично и доказательно раскрывает проблему, предложенную в билете; обучающийся демонстрирует умение вести диалог и вступать в научную дискуссию

4	ОПК-5З.1, ОПК-5У.1, ОПК-5В.1, ПК-6.З.1,ПК-6.У.1,ПК-6.В.1	А-III/6-1.5.		Текущий контроль	Лабораторная работа	Защита лабораторных работ по дисциплине "Информатика", По результатам выполнения работ создается отчет. Образец отчета прилагается.	Работа выполнена не полностью и объем выполненной части работы не позволяет сделать правильных выводов: если опыты, измерения, вычисления, наблюдения производились неправильно	Работа выполнена не полностью, но объем выполненной части позволяет получить правильные результаты и выводы, если в ходе проведения опыта, измерений, вычислений и наблюдений были допущены ошибки	Работа выполнена в полном объеме с соблюдением необходимой последовательности проведения опытов, измерений, вычислений и наблюдений; все опыты проводит в условиях и режимах, обеспечивающих получение правильных результатов и выводов; в отчете правильно и аккуратно выполняет все записи, таблицы, рисунки, чертежи, графики, вычисления; но допускает несколько недочетов, оформление отчета согласно установленным требованиям; при защите отчета обучающийся ответил в основном на все контрольные вопросы	Работа выполнена в полном объеме с соблюдением необходимой последовательности проведения опытов, измерений, вычислений и наблюдений; все опыты проводит в условиях и режимах, обеспечивающих получение правильных результатов и выводов; в отчете правильно и аккуратно выполняет все записи, таблицы, рисунки, чертежи, графики, вычисления; оформление отчета согласно установленным требованиям; при защите отчета обучающийся ответил на все контрольные вопросы

5	ОПК-5З.1, ОПК-5У.1, ОПК-5В.1, ПК-6.З.1,ПК-6.У.1,ПК-6.В.1	А-III/6-1.5.	4.1.7,5.1,5.2,5.3,5.4,5.5,5.6,6.1,6.2,6.3,6.4,6.5,6.6,7,9,8	Текущий контроль	Лабораторная работа	Защита лабораторных работ по дисциплине "Информатика",	Работа выполнена не полностью и объем выполненной части работы не позволяет сделать правильных выводов: если опыты, измерения, вычисления, наблюдения производились неправильно	Работа выполнена не полностью, но объем выполненной части позволяет получить правильные результаты и выводы, если в ходе проведения опыта, измерений, вычислений и наблюдений были допущены ошибки	Работа выполнена в полном объеме с соблюдением необходимой последовательности проведения опытов, измерений, вычислений и наблюдений; все опыты проводит в условиях и режимах, обеспечивающих получение правильных результатов и выводов; в отчете правильно и аккуратно выполняет все записи, таблицы, рисунки, чертежи, графики, вычисления; но допускает несколько недочетов, оформление отчета согласно установленным требованиям; при защите отчета обучающийся ответил в основном на все контрольные вопросы	Работа выполнена в полном объеме с соблюдением необходимой последовательности проведения опытов, измерений, вычислений и наблюдений; все опыты проводит в условиях и режимах, обеспечивающих получение правильных результатов и выводов; в отчете правильно и аккуратно выполняет все записи, таблицы, рисунки, чертежи, графики, вычисления; оформление отчета согласно установленным требованиям; при защите отчета обучающийся ответил на все контрольные вопросы
6	ОПК-5З.1, ОПК-5У.1, ОПК-5В.1, ПК-6.З.1,ПК-6.У.1,ПК-6.В.1	А-III/6-1.5.	4.1.7,5.1,5.2,5.3,5.4,5.5,5.6,6.1,6.2,6.3,6.4,6.5,6.6,7,9,8	Промежуточная аттестация	Итоговый тест	Итоговый тест (20 вопросов) письменно, 60 минут	0-49% правильных ответов	50-69% правильных ответов	70-89% правильных ответов	90-100% правильных ответов

Федеральное агентство морского и речного транспорта
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Волжский государственный университет водного транспорта»

Кафедра информатики, систем управления и телекоммуникаций

Серия «Информационные технологии
в системах управления и телекоммуникации»

Выпуск 20

Р.Н. Гурьяшова

Лабораторный практикум по информатике

Учебное пособие
для студентов очного и заочного обучения
технических специальностей

Рекомендовано
федеральным государственным бюджетным образовательным учреждением высшего профессионального образования «Московский государственный технический университет имени Н.Э. Баумана» в качестве учебного пособия для студентов высших учебных заведений, обучающихся по специальности 180407.65 «Эксплуатация судового оборудования и средств автоматики» и направлениям подготовки 180100.62 «Кораблестроение, океанотехника и системотехника объектов морской инфраструктуры»; 190600.62 «Эксплуатация транспортно-технологических машин и комплексов»

Рег. номер рецензии 2973 от 02.02.2015 г. МГУП

Нижний Новгород
Издательство ФГБОУ ВО «ВГУВТ»
2015

УДК 004.42
Г95

Редакционная коллегия серии «Информационные технологии в системах управления и телекоммуникаций»:
д. т. н., профессор *Ю.С. Федосенко* (отв. редактор),
д. т. н., профессор *М.М. Чиркова*,
к. т. н., доцент *В.И. Логинов*,
к. т. н., доцент *А.В. Преображенский*

Гурьяшова, Р.Н.
Лабораторный практикум по информатике : учеб. пособие для студ. оч. и заоч. обуч. технич. специальностей / Р.Н. Гурьяшова. – Н. Новгород : Изд-во ФГБОУ ВО «ВГУВТ», 2015. – 144 с.

Цель пособия – формирование у студентов начальных навыков алгоритмизации и программирования, составляющих основу профессиональной деятельности современного инженера. Методическое изложение материала основано на многолетнем опыте преподавания дисциплины «Информатика» студентам технических специальностей в ВГАВТ.

Излагаются основные типовые приёмы решения задач и структуры базовых алгоритмов. Алгоритмы приводятся как в словесном описании, так и в виде схем. Приводятся также листинги программ, иллюстрирующие выполнение лабораторных работ на императивных языках программирования высокого уровня – C, Fortran и Basic.

Для студентов, обучающихся по техническим специальностям и направлениям подготовки в высших учебных заведениях водного транспорта.

Работа рекомендована к изданию кафедрой информатики, систем управления и телекоммуникации (протокол № 5 от 15.05.2014).

© ФГБОУ ВО «ВГУВТ», 2015

Введение

«Единственный способ изучать новый язык программирования – писать на нём программы»

Брайэн Керниган

В инженерной практике необходимо умение ставить задачи, разрабатывать алгоритмы их решения, выполнять расчёты, а также анализировать полученные результаты. Поэтому инженер должен грамотно применять персональный компьютер для решения различных возникающих в профессиональной деятельности задач.

В пособии рассматриваются типовые приёмы и структуры линейных, разветвлённых и циклических алгоритмов. Для иллюстрации алгоритмов приводятся следующие лабораторные работы:

- **Линейный алгоритм**
  - «Вычисление значения выражения» (1 лаб. работа)
- **Разветвлённый алгоритм**
  - «Вычисление значения величины, заданной ветвлением» (2)
- **Циклический алгоритм**
  - «Табулирование функции одной переменной» (3)
  - «Табулирование функции двух переменных» (2)
  - «Одномерный массив» (6)
  - «Двумерный массив – матрица» (5)

Для каждой лабораторной работы приводятся листинги программ на языках программирования высокого уровня – C, Fortran, Basic, используемых в учебном процессе. Приведённые программы с большим количеством комментариев, поясняющих программный код, послужат студентам справкой при выполнении заданий.

Кроме того, по каждой теме приведены контрольные задания для проверки знаний, а также задания для самостоятельной работы.

Тема «Сегментация программ» в пособии не рассматривается (см. стр.140, п.5).

Все программы тестировались в средах Microsoft Visual Studio C, QBasic 1.0 (1.1), GNU g95 Fortran compiler. Программы могут быть легко адаптированы для других версий языков.

Цель пособия – научить студентов пользоваться инструментарием языков программирования при решении задач.

1. Рекомендации к лабораторной работе

Для выполнения лабораторной работы необходимо:

1. В соответствии с заданием и типом алгоритма ввести обозначения переменных. Например, сумму, произведение, количество, среднее арифметическое, наибольшее, наименьшее значение с учётом смысла удобнее обозначить как: S, P, K, SR, MAX, MIN.
2. Имена всех переменных, используемых в задаче, свести в таблицу с пояснением их смысла.
3. Выполнить ручной расчет результатов. Задания подобраны так, что ручной расчёт не является трудоёмким.
4. Разработать схему алгоритма решения задачи.
5. Составить макет печати исходных данных и результатов, предусмотрев при этом поясняющие текстовые заголовки.
6. В соответствии со схемой и макетом печати написать программу. Для повышения её читаемости рекомендуется грамотное использование:

- отступов при записи программного кода (правило рельефа);
- осмысленных и кратких имён переменных;
- комментариев, поясняющих программный код.

7. Отладить и выполнить программу.
8. Проанализировать полученные результаты. Ручной и машинный расчёты должны совпадать.
9. Представить отчёт-презентацию лабораторной работы.

Презентация состоит из следующих разделов:

1. Постановка задачи.
2. Таблица имён переменных с пояснением их смысла.
3. Ручной расчет результатов (контрольных точек).
4. Схема алгоритма и программа решения задачи.
5. Распечатка (макет печати) исходных данных и результатов.

В качестве примера оформления приведён отчёт для работы 1 (стр.137).

2. Примеры выполнения лабораторных работ

Лабораторная работа 1

Вычисление значения выражения

Тема. Линейный алгоритм

Индивидуальные задания – табл. 3.1 (стр.92)
Математические функции – табл. 5.1 (стр.136)

1. Задание

Составить программу решения следующей задачи.
Вычислить значение величины Z:

$$Z = (\ln|b| + \cos a)^2 - \frac{\sqrt{\ln 3 + \cos b}}{a\ (\ln a + \cos\ 2^\circ)},\ \text{где}\ a = 1{,}5 \quad b = -2.$$

2. Решение

Для удобства отладки программы введем обозначения:

$R = \ln|B| + \cos A$; $W = \ln 3 + \cos B$; $Q = \ln A + \cos 2^\circ$.

Тогда вычисление Z можно представить в виде:

$$Z = R^2 - \frac{\sqrt{W}}{A \cdot Q}$$

Обозначения переменных			
Имя переменной	Тип переменной		Пояснение
	Fortran	**C**	
A, B	**REAL**	**FLOAT**	Исходные данные
R, W, Q, Z	**REAL**	**FLOAT**	Рабочие переменные и результат

Примечание. *У тригонометрических функций аргумент указывается в радианах. Градусы переводятся в радианы по формуле* $\frac{x \cdot \pi}{180}$. *Например, cos 2º следует записать как* ***COS(2*3.1415/180)***.

3. Алгоритм решения задачи

Алгоритм имеет линейную структуру. Представим словесное описание и схему алгоритма (рис. 2.1):

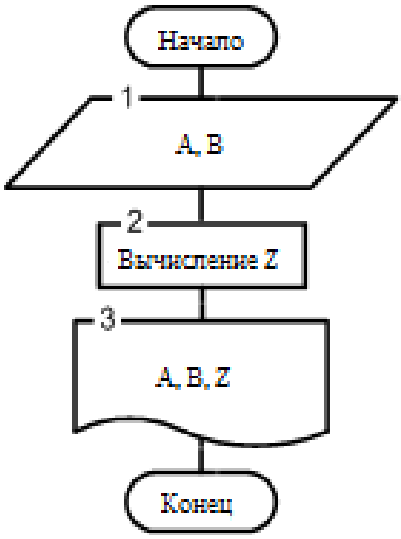


Рис. 2.1. Схема алгоритма к лабораторной работе 1 «Вычисление значения выражения»

Алгоритм состоит из следующих действий:

1) Ввод данных – значений A, B;

2) Вычисление результата Z;

3) Вывод значений A, B, Z;

4) Конец.

Приведем два варианта С-программы: с функцией и без использования функции. Выделим в функцию с именем **f** повторяющееся арифметическое выражение вида

$$\ln x + \cos y$$

В процессе вычисления значения Z обращение к функции **f** происходит три раза, а именно:

Имя переменной	Выражение	Вызов функции F
R	ln |B| + cos A	F(|B|, A)
W	ln 3 + cos B	F(3, B)
Q	ln A + cos 2º	F(A, 2º)

4. Листинг С-программы

Внимание. *При вводе программы следует иметь в виду, что С различает регистр букв, т.е., «**x**» и «**X**» – это две разные переменные.*

Без функции	С функцией
Программный код	

`#include<stdio.h>`
`#include<conio.h>`
`#include<math.h>`
`void main()`
`{`
`// Объявление типа данных`
	`#include<stdio.h>`
	`#include<conio.h>`
	`#include<math.h>`
	`// Прототип функции F`
	`float f(float x, float y);`
	`void main()`

```
double a = 1.5,b = -2;
const double pi = 3.1415;
double r,w,q,z;
// Вычисление R,W,Q,Z
r = log(abs(b))+ cos(a);
w = log(3.0)+ cos(b);
q = log(a)+cos(2*pi/180);
z = r*r - sqrt(w)/(a*q);
// Вывод A,B,Z
printf(" A = %.1lf",a);
printf(" B = %.0lf\n",b);
printf(" Z = %.4lf",z);
getch();
return;
}
```

```
{
// Объявление типа данных
float a = 1.5,b = -2;
const float pi = 3.1415;
float r,w,q,z;
// Вызов функции F
r = f(abs(b),a);
w = f(3.0,b);
q = f(a,2*pi/180);
z = r*r - sqrt(w)/(a*q);
// Вывод A,B,Z
printf(" A = %.1f ",a);
printf(" B = %.0f\n ",b);
printf(" Z = %.4f ",z);
getch();
return;
}
// Функция F
float f(float x,float y)
{
return(log(x)+cos(y));
}
```

5. Листинг Fortran и Basic-программ

Программный код	
Fortran	**Basic**

```
! Объявление данных
REAL :: A = 1.5 ; B = -2
REAL :: PI = 3.1415
REAL :: R,W,Q,Z
! Вычисление R,W,Q,Z
R = LOG(ABS(B))+ COS(A)
W = LOG(3.0)+ COS(B)
Q = LOG(A)+ COS(2*PI/180)
Z = R**2 - SQRT(W)/(A*Q)
! Вывод значений A,B,Z
PRINT 1, A,B,Z
1 FORMAT(1X,' A = ',  &
F3.1,' B = ',F3.0/    &
1X,'Z = ',E11.4 )
END
```

```
CLS
' Исходные данные
A = 1.5: B = -2
PI = 3.1415
' Вычисление R,W,Q,Z
R = LOG(ABS(B))+ COS(A)
W = LOG(3) + COS(B)
Q = LOG(A)+ COS(2*PI/180)
Z = R^2 - SQR(W)/(A*Q)
' Вывод значений A,B,Z
PRINT " A = ";A;
PRINT " B = ";B
PRINT " Z = ";Z
END
```

6. Макет печати результатов

По С-программе

```
A = 1.5 B = -2
Z = 0.1915
```

По Fortran-программе

```
 A = 1.5 B = -2.
 Z = 0.1915E+00
```

Лабораторная работа 2

Вычисление значения величины, заданной в форме ветвления

Тема. Разветвлённый алгоритм

Индивидуальные задания – табл. 3.2 (стр.94)
Математические функции – табл. 5.1 (стр.136)

1. Задание

Составить схему алгоритма и программу для следующей задачи.
Вычислить значение величины Z, заданной ветвлением вида:

$$Z=\begin{cases} a\cdot\min(2\cdot b,k) & \text{при } a<b \text{ и } a=1\\ a\cdot b-k^3 & \text{при } a<b \text{ и } a\neq 1\\ a^2+\frac{b}{k} & \text{при } a\geq b \text{ и } b\geq 0\\ \sqrt{|b|}+a\cdot k & \text{при } a\geq b \text{ и } b<0 \end{cases}$$

Расчёт значений Z приведён в следующей таблице:

Исходные данные			Формула для Z	Значение Z
a	b	k		
1	3	4	$a\cdot\min(2b,k)$	4
2	4	1	$a\ b\quad k^3$	7
3	2	1	a^2+b/k	11
1	– 4	3	$\sqrt{\lvert b\rvert}+a\cdot k$	5

2. Решение

Обозначения переменных			
Имя переменной	Тип переменной		Пояснение
	Fortran	C	
A, B, K	REAL	FLOAT	Исходные данные
Z	REAL	FLOAT	Искомая величина,результат

Алгоритм имеет разветвленную структуру и содержит пять ветвей (рис. 2.2). Каждая ветвь соответствует одному из условий задачи и определяет путь («маршрут») следования вычислительного процесса в соответствии с проверяемым условием. «Маршруты» для каждой из ветвей приведены в следующей таблице.

Условие	**Формула для Z**	**<Маршрут> Номера блоков в схеме алгоритма**
A < B и A = 1, 2B < K	$a \cdot \min(z \cdot b, k)$	1 – 2 – 3 – 4 – 5 – 11
A < B и A = 1, 2B ≥ K		1 – 2 – 3 – 4 – 6 – 11
A < B и A ≠ 1	$a \cdot b - k^3$	1 – 2 – 3 – 7 – 11
A ≥ B и B ≥ 0	$a^2 + b/k$	1 – 2 – 8 – 10 – 11
A ≥ B и B < 0	$\sqrt{\lvert b \rvert} + a \cdot k$	1 – 2 – 8 – 9 – 11

Схема алгоритма представлена на рис. 2.2:

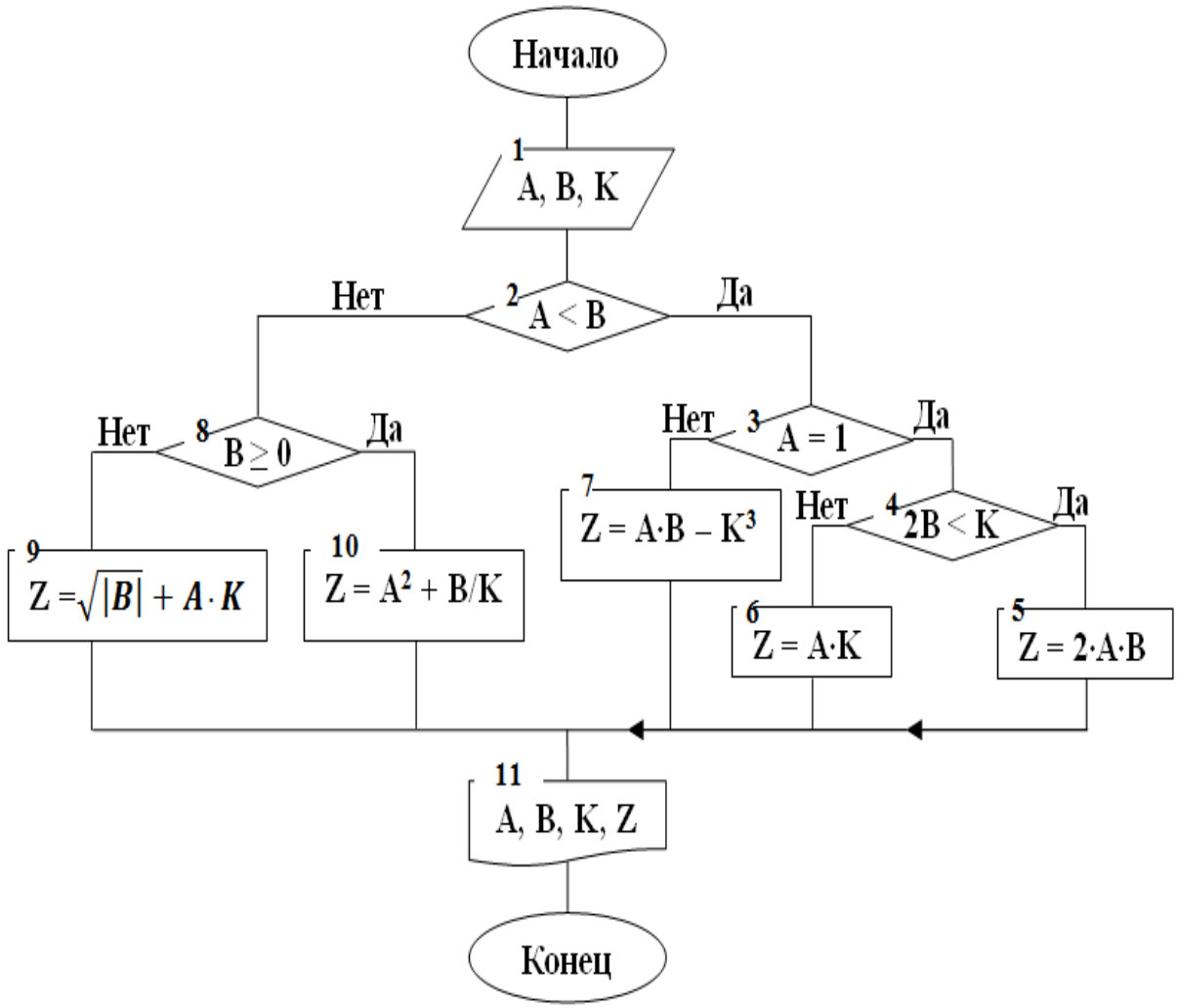


Рис. 2.2 Схема алгоритма к лабораторной работе 2

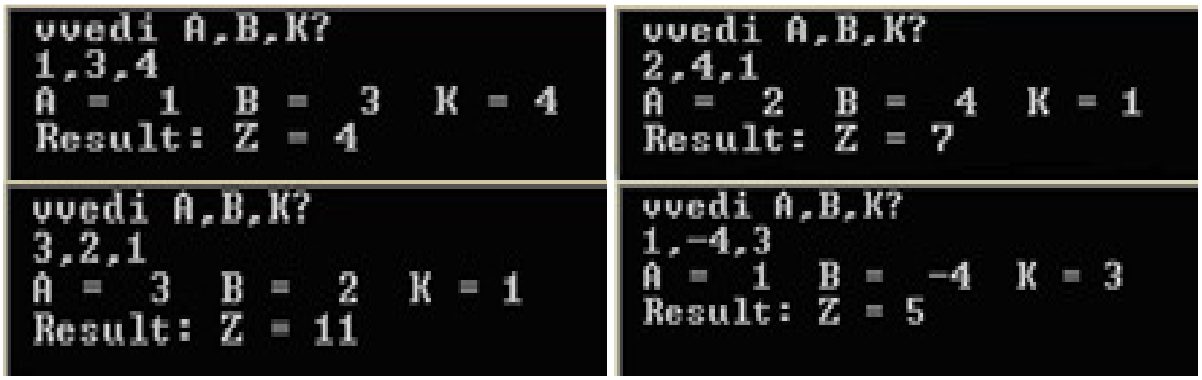
«Вычисление значения величины, заданной в форме ветвления»

3. Листинг С-программы «Разветвлённый алгоритм»

Внимание. *При вводе программы следует иметь в виду, что С различает регистр букв, т.е., «**x**» и «**X**» – это две разные переменные.*

```
#include<stdio.h> // Файлы заголовков
#include<conio.h>
#include<math.h>
void main()
{
float a,b,k,z; // Объявление типа данных
// Ввод данных с клавиатуры по запросу с экрана,
// числа при вводе разделяются запятой
 printf(" vvedi A,B,K? \n ");
 scanf("%f,%f,%f",&a,&b,&k);
 printf(" A = %.0f B = %.0f K = %.0f \n",a,b,k);
if(a<b) // Если A < B
   if(a == 1) // Если A = 1
      if(2*b < k) z = 2*b*a; // Вычисление MIN(2·B,K)
      else   z = a*k;
   else // Иначе,если A ≠ 1
      z = a*b - pow(k,3);
else // Иначе,если A >= B
   if(b >= 0) // Если B >= 0
      z = a*a + b/k;
   else // Иначе,если B < 0
      z = sqrt(abs(b)) + a*k;
 printf(" Result: Z = %.0f ",z); // Вывод Z
 getch();
 return;
}
```

4. Макет печати (по С-программе)



Как видно, результаты ручного и машинного расчётов совпадают.

5. Листинг Basic-программы «Разветвлённый алгоритм»

Программный код

```
INPUT "Введи A, B, K"; A, B, K
IF A < B THEN ' Если A < B
  IF A = 1 THEN ' Если A = 1
    IF 2 * B < K THEN Z = 2 * B * A ELSE Z = K * A
  ELSE ' Иначе, если A ≠ 1
         Z = A * B - K ^ 3
  END IF
ELSE       ' Иначе, если A ≥ B
  IF B >= 0 THEN ' Если B >= 0
         Z = A ^ 2 + B / K
  ELSE  ' Иначе, если B < 0
         Z = SQR(ABS(B)) + A * K
  END IF
END IF
PRINT " A = "; A; " B = "; B; " C = "; C
PRINT USING " РЕЗУЛЬТАТ Z = ## "; Z
END
```

Результаты приведены в макете печати в п.4.

Лабораторная работа 3

Вычисление значения величины, заданной в форме ветвления

Тема. Разветвлённый алгоритм

Индивидуальные задания – табл. 3.3 (стр.97)

Математические функции – табл. 5.1 (стр.136)

1. Задание

Составить схему алгоритма и программу для следующей задачи.

Вычислить значение величины Z, заданной ветвлением вида:

$$Z = \begin{cases} \min\{a, b \cdot \max(2a, c^2)\} & \text{при} \quad c > b \\ c + \min(a, b) & \text{при} \quad c \le b \text{ и } a \ne 1 \\ c \cdot \max\left(2b, \sqrt{|a|}\right) & \text{при} \quad c \le b \text{ и } a = 1 \end{cases}$$

Расчёт значений Z приведён в следующей таблице:

Исходные данные			Формула для Z	Значение Z
a	b	c		
1	2	3	$\min\{a, b \cdot \max(2a, c^2)\}$	1
4	3	1	$c + \min(a, b)$	4
1	2	1	$c \cdot \max\left(2b, \sqrt{\lvert a \rvert}\right)$	4

2. Решение

Обозначения переменных			
Имя переменной	Тип переменной		Пояснение
	Fortran	**C**	
A, B, C	**REAL**	**FLOAT**	Исходные данные
Z	**REAL**	**FLOAT**	Искомая величина, результат

Алгоритм имеет разветвленную структуру и содержит восемь ветвей. В соответствии со схемой (рис. 2.3) необходимо *самостоятельно* представить все его ветви (см. лаб. работу 2, стр.9).

Схема алгоритма решения задачи представлена на рис. 2.3:

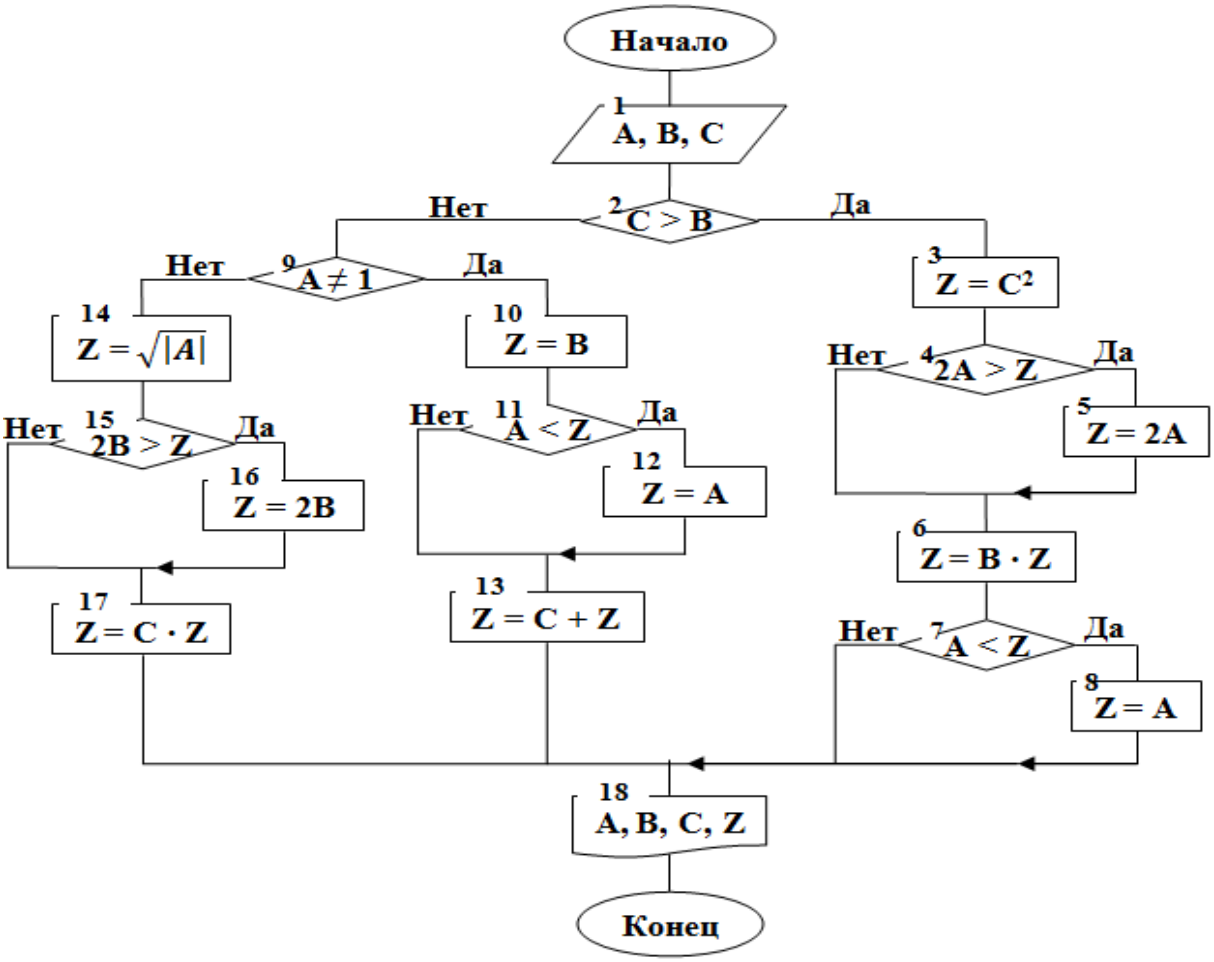


Рис. 2.3 Схема алгоритма к лабораторной работе 3

3. Листинг С-программы «Разветвлённый алгоритм»

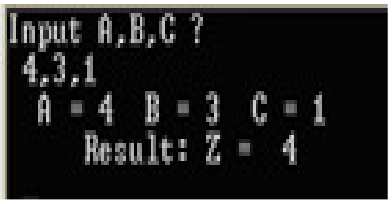
Внимание. *При вводе программы следует иметь в виду, что С различает регистр букв, т.е., «**x**» и «**X**» – это две разные переменные.*

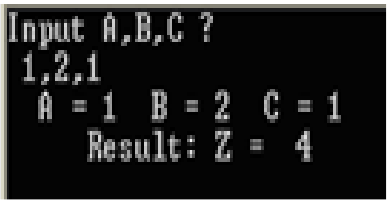
```
#include<stdio.h> // Файлы заголовков
#include<conio.h>
#include<math.h>
void main()
{
float a,b,c,z; // Объявление типа данных
// Ввод данных с клавиатуры по запросу с экрана,
// числа при вводе разделяются запятой
 printf(" Input A,B,C ?\n ");
 scanf("%f,%f,%f",&a,&b,&c);
//  Вычисление  Z = MIN(A,B*MAX(2*A,C*C))
 if(c > b)  // Если C > B
   {
    z = c*c;
    if(2*a > z) z = 2*a;
    z = b*z;
    if (a < z) z = a;
   }
 else //  Иначе,если C <= B
// Вычисление  Z = C + MIN(A,B)
    if(a != 1) // Если A ≠ 1
       {
        z = b;
        if (a < z) z = a;
        z = c + z;
       }
    else // Иначе,если A = 1
// Вычисление Z = C*MAX(2*B,√|A|)
       {
        z = sqrt(abs(a));
        if (2*b > z) z = 2*b;
        z = c*z;
       }
// Вывод данных A, B, C и результата Z
printf(" A = %.0f  B = %.0f  C = %.0f \n ",a,b,c);
printf(" Result: Z =  %.0f ",z);
getch();
return;
```

```
}
```

4. Макет печати (по С-программе)







Как видно, результаты ручного и машинного расчётов совпадают.

5. Листинг Fortran-программы

Программный код

```
! Разветвленный алгоритм
PROGRAM VETVI3
REAL :: A,B,C,Z ! Объявление типа данных
! Ввод данных по запросу с экрана
! числа при вводе разделяются пробелом
PRINT *,' Введите A,B,C ?'
READ(*,*)A,B,C
IF(C > B)THEN          !  Если  C > B
! Вычисление Z = MIN(A,B*MAX(2*A,C**2))
   Z = C ** 2
   IF(2*A > Z) Z = 2*A
   Z = B * Z
   IF(A < Z) Z = A
ELSE                   ! Иначе если C <= B
! Вычисление Z = C + MIN(A,B)
     IF(A /= 1)THEN  ! Если A ≠ 1
        Z = B
        IF(A < Z) Z = A
        Z = C + Z
     ELSE              ! Иначе, если A = 1
! Вычисление Z = C*MAX(2*B, √|A|)
        Z = SQRT(ABS(A))
        IF(2*B > Z) Z = 2*B
        Z = C * Z
     ENDIF
ENDIF
! Вывод данных A, B, C и результата Z
WRITE (*, 3) A,B,C,Z
3 FORMAT(2X,' A = ',F3.0,' B = ',F3.0,      &
   ' C = ',F3.0/ ' РЕЗУЛЬТАТ Z = ',F3.0)
END
```

Результаты приведены в макете печати в п.4.

Контрольные задания для проверки знаний

Тема. Разветвлённый алгоритм

Для п.1-3: Исходные данные и результат представлены в таблице. При несовпадении результата необходимо исправить ошибки и повторно выполнить программу.

Составить схему алгоритма и программу для следующих задач:

1. Вычислить значение сложной функции Y = F(X).

$$Y = \begin{cases} X + 4 & \text{при} \quad X < -6 \\ 2 \cdot X + 10 & \text{при} \quad -6 \le X < 0 \\ 10 - X & \text{при} \quad 0 \le X < 5 \\ 5 & \text{при} \quad X \ge 5 \end{cases}$$

X	Y
–8	–4
–1	8
4	6
7	5

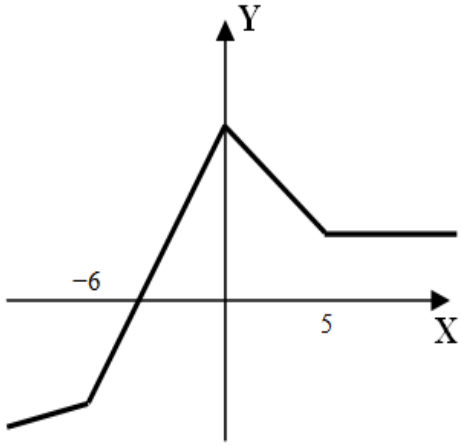


Рис. 2.4 «Вид функции Y = F(X)»

Для проверки программу необходимо выполнить четыре раза для каждого значения аргумента X.

2. Вычислить значение величины Z, заданной ветвлением.

$$Z = \begin{cases} \max(A; 3 \cdot B) & \text{при } A \ge B \\ \min(2 \cdot A; B^2) & \text{при } A \le B \end{cases}$$

Данные		Результат
A	B	Z
4	1	4
1	3	2

Для проверки программу необходимо выполнить дважды.

3. Вычислить значение сложной функции Y = F(X).

$$Y = \begin{cases} X^2 + 1 & \text{при} \quad X < -1 \\ \sqrt{|X|} & \text{при} \quad -1 \le X \le 1 \\ X^3 - 2 & \text{при} \quad X > 1 \end{cases}$$

X	Y
–2	5
–1	1
2	6

Для проверки программу необходимо выполнить три раза, для каждого из заданных значений X.

4. Вычислить значение величины $Z = C \cdot \min\{2 \cdot A, \max(A^2, C), B^3\}$. Исходные данные задать самостоятельно.

5. Определить попадание точки (X,Y) в область, ограниченную параболой $F(X) = -X^2 + 4$ и прямой $G(X) = 0$.

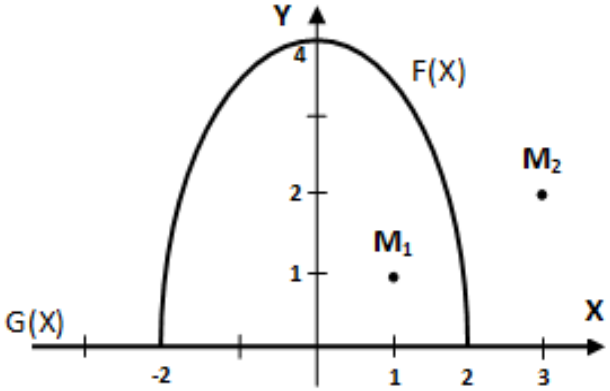


Рис. 2.5 «Вид области»

Необходимо в соответствии с графиком записать условие принадлежности точки (X,Y) заданной области в виде ветвления.

Для проверки программу следует выполнить для двух точек, например, M_1 (1;1) и M_2 (3;2). Согласно рис. 2.5, точка M_1 (1;1) попадает в заданную область, а точка M_2 (3;2) – нет.

6. Определить принадлежность точки (X,Y) области, ограниченной прямыми X = –2, X = 2, Y = –2, Y = X и Y = –X. Область представлена на рис. 2.6.

Необходимо в соответствии с графиком записать условие принадлежности точки (X,Y) заданной области в виде ветвления.

Для проверки программу следует выполнить для двух точек, например, M_1 (1;–1) и M_2 (0;2). Согласно рис. 2.6, точка M_1 (1;–1) лежит внутри заданной области, а точка M_2 (0;2) – вне её.

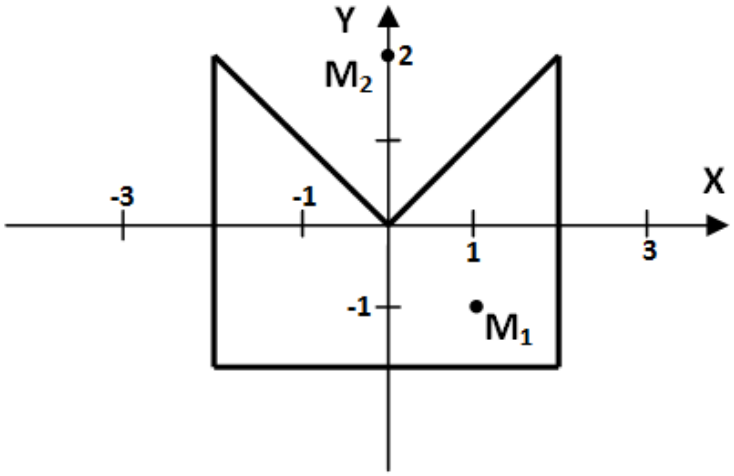


Рис. 2.6 «Вид области»

Лабораторная работа 4

Исследование функции одной переменной

Тема. Табулирование функции одной переменной

Индивидуальные задания – табл. 3.4 (стр.100)
Математические функции – табл. 5.1 (стр.136)

1. Задание

Составить программу решения следующей задачи.

Вычислить таблицу значений аргумента **X** и функции **Y = F(X)** при условии, что аргумент **X** изменяется в интервале $\mathbf{X_{min} \le X \le X_{max}}$ с шагом Δ**X**. **Протабулировать функцию и найти наибольшее из её отрицательных значений и соответствующее значение аргумента X для функции вида**

$$F(x) = \sqrt[3]{2x} - e^{-x} \cdot tg^2(3x)$$

Исходные данные

Начальное значение аргумента	Конечное значение аргумента	Шаг изменения аргумента
$\mathbf{X_{min}}$	$\mathbf{X_{max}}$	ΔX
0,6	0,8	0,05

2. Решение

Обозначим: $\mathbf{Y_{max}}$, $\mathbf{X_m}$ – наибольшее из отрицательных значений функции и соответствующее ему значение аргумента X.

Обозначения переменных				
Имя переменной		Тип переменной		Пояснения
в задаче	в программе	**Fortran**	**C**	
$\mathbf{X_{min}}$, $\mathbf{X_{max}}$	XMIN, XMAX	**REAL**	**DOUBLE**	Начальное и конечное значение аргумента X
ΔX	DX	**REAL**	**DOUBLE**	Шаг изменения аргумента X
X, Y	X, Y	**REAL**	**DOUBLE**	Значение аргумента и функции
$\mathbf{Y_{max}}$, $\mathbf{X_m}$	YMAX, XM	**REAL**	**DOUBLE**	Наибольшее из отрицательных значений функции и соответствующее значение аргумента X

3. Ручной расчёт результатов

Вычислим значение F(X) в начальной и конечной точках.

Контрольные точки	Значение X	Значение Y
X_{min}	0,6	–9.02
X_{max}	0,8	0.793

Результаты можно представить графически и в виде таблицы.

Графический способ

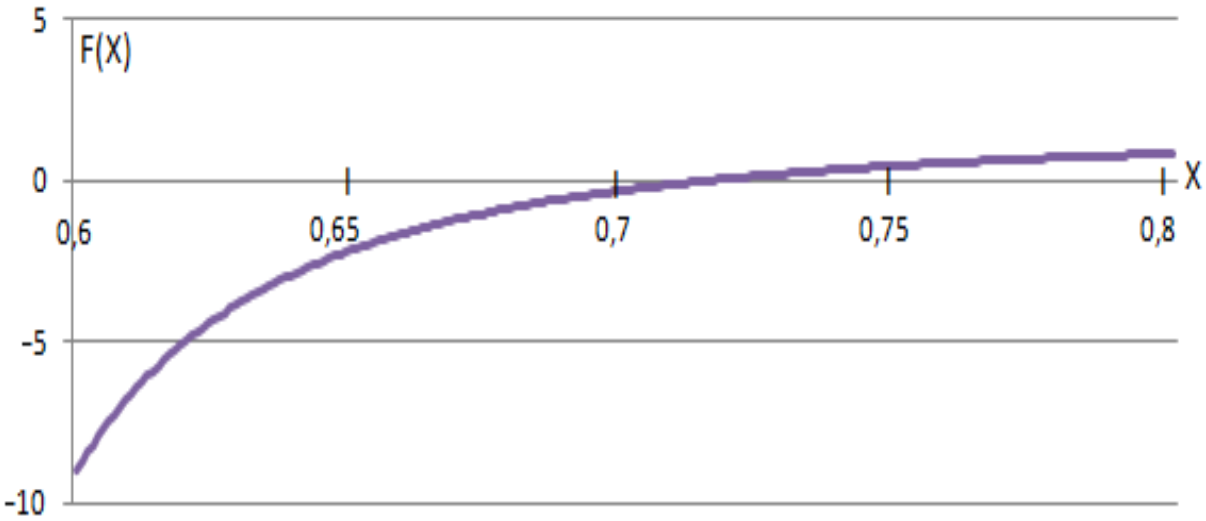


Таблица значений аргумента X и функции Y = F(X)

X	Y
0,6	-9,02

Количество значений (x;y) [строк в таблице]

0,65	-2,196
0,7	**-0,333**
0,75	0,42
0,8	0,793

определяется как:

$$\frac{Xmax - Xmin}{\Delta X} + 1 = \frac{0,8-0,6}{0,05} + 1 = 5$$

Наибольшее из отрицательных значений функции $\mathbf{Y_{max} = -0,333}$ достигается при $\mathbf{X_m = 0,7}$.

Для решения задачи используется циклический алгоритм табулирования функции и вычисления наибольшего значения.

4. Алгоритм

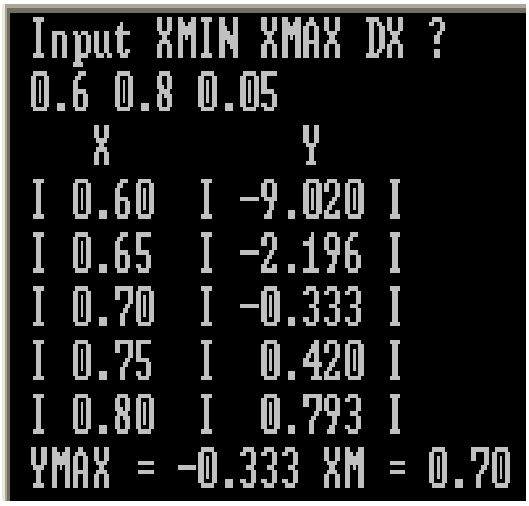
Алгоритм состоит из следующих действий:

1. Ввод исходных данных – значений переменных X_{min} , X_{max} , ΔX.
2. Задание начального значения Ymax= –1.0E38 (заведомо малого числа -10^{38}).
3. *Начало цикла по X.*
4. Вычисление значения функции Y = F(X).
5. Вывод текущего значения аргумента X и функции Y.
6. Проверка условия Y < 0. Если условие выполняется – перейти к п.7, иначе – к п.9.
7. Проверка условия $Y > Y_{max}$. Если условие выполняется – перейти к п.8, иначе – к п.9.
8. Присвоить значения переменным: $Y_{max} = Y$, $X_m = X$.
9. *Конец цикла по X.*
10. Вывод результатов – значений Y_{max}, X_m.
11. Конец.

Схема алгоритма представлена на рис. 2.7 (стр.21).

5. Макет печати результатов

По С-программе **По Basic-программе**




```
      ГРАНИЦЫ ИНТЕРВАЛА
        0.6 <= X <= 0.8
        ШАГ DX = 0.05
           ТАБЛИЦА
  X = 0.60    Y = -9.020E+00
  X = 0.65    Y = -2.196E+00
  X = 0.70    Y = -3.331E-01
  X = 0.75    Y =  4.200E-01
  X = 0.80    Y =  7.925E-01
YMAX =-3.331E-01 при X = 0.70
```

Как видно, результаты ручного и машинного расчётов совпадают.

6. Листинг С-программы «Табулирование функции»

Внимание. *При вводе программы следует иметь в виду, что С различает регистр букв, т.е., «**x**» и «**X**» – это две разные переменные.*

*Условия **Y < 0** и **Y > YMAX** объединены логической связкой **&&**.*

```
#include<stdio.h> // Файлы заголовков
#include<conio.h>
#include<math.h>
    void main()
{
double xmin,xmax,dx; // Объявление типа данных
double x,y,xm,ymax = -1.0e38;
// Ввод данных с клавиатуры по запросу с экрана,
// числа при вводе разделяются пробелом
printf(" Input XMIN XMAX DX ?\n ");
scanf("%f %f %f",&xmin,&xmax,&dx);
printf("    X          Y \n"); // Вывод заголовка
for(x = xmin;x<=xmax + dx/3;x+=dx) // Цикл по X
  {
y = pow((2.*x),(1./3.))-exp(-x)*pow(tan(3*x),2);
// Вывод значений аргумента X и функции Y
printf(" I %.2lf  I  %6.3lf I \n ",x,y);
if(y < 0 && y > ymax) // Если Y < 0 и Y > YMAX
```

```
    {
     ymax = y; xm = x;
    }
  }
// Вывод значений YMAX, XM
printf(" YMAX = %.3lf  XM = %.2lf ",ymax,xm);
getch();
return;
}
```

Примечание. При выполнении оператора **for(x = xmin;x<=xmax;x+=dx)** в результате накопления ошибки может произойти «потеря» значения функции при $\mathbf{X = X_{max}}$. Поэтому значение **xmax** следует увеличить:
for(x = xmin;x<=xmax + dx/3;x+=dx).

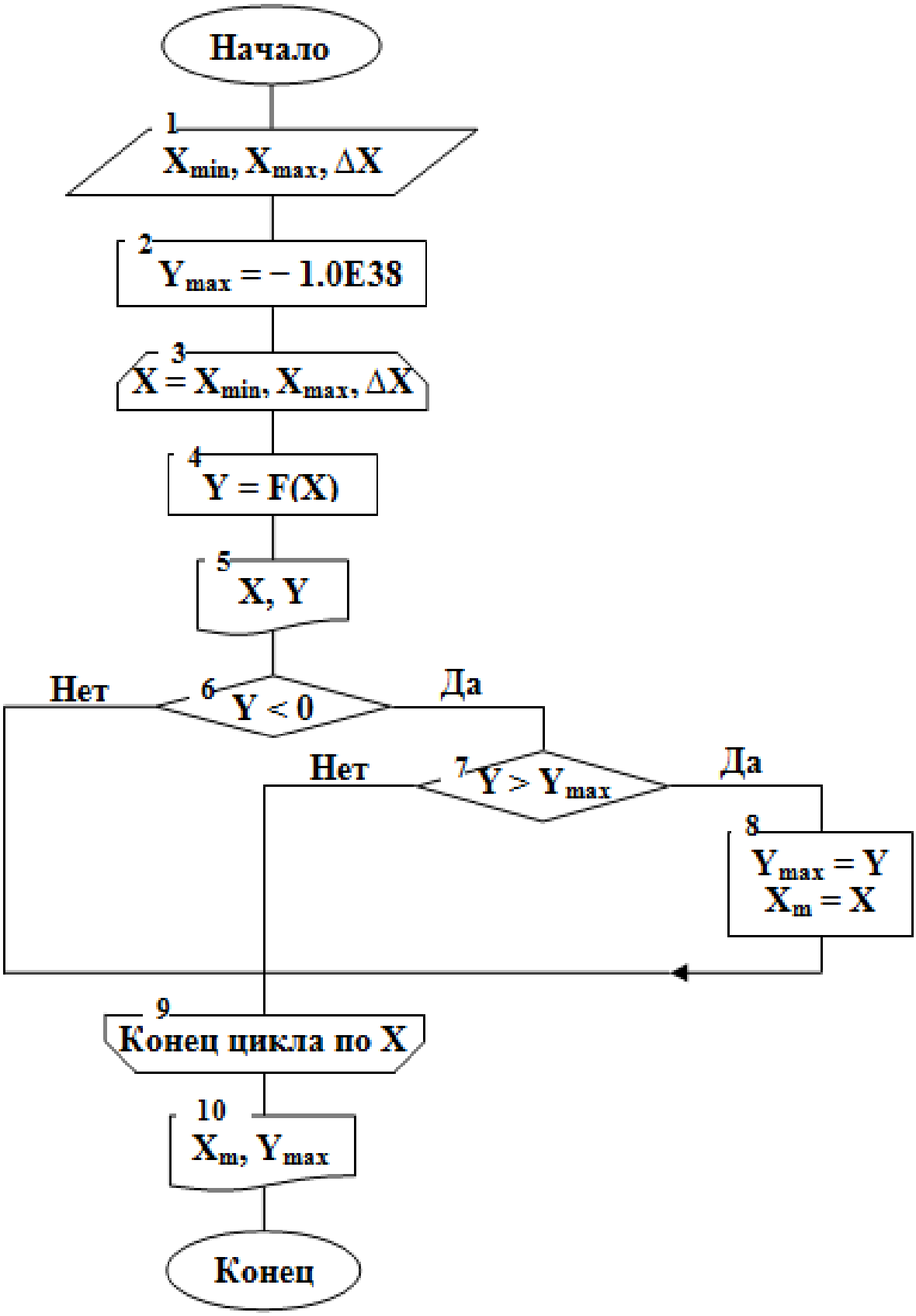


Рис. 2.7 Схема алгоритма к лабораторной работе 4 «Исследование функции одной переменной»

7. Листинг Basic-программы «Табулирование функции»

*Условия **Y < 0** и **Y > YMAX** объединены логической связкой **AND**.*

```
CLS
READ XMIN, XMAX, DX ' Ввод исходных данных
DATA 0.6,0.8,0.05
' Вывод исходных данных
PRINT TAB(5); " ГРАНИЦЫ ИНТЕРВАЛА "
PRINT TAB(6); USING " #.# <= X <= #.# "; XMIN; XMAX
PRINT TAB(8); USING " ШАГ DX = #.## "; DX
PRINT TAB(10); " ТАБЛИЦА " ' Вывод заголовка
YMAX = -1E+38
FOR X = XMIN TO XMAX + DX / 3 STEP DX ' Цикл по X
  Y = (2 * X) ^ (1 / 3) - EXP(-X) * TAN(3 * X) ^ 2
  PRINT USING " X = #.## Y = ##.###^^^^ "; X; Y
  IF Y < 0 AND Y > YMAX THEN ' Если Y < 0 и Y > YMAX
    YMAX = Y: XM = X
  END IF
NEXT X ' Конец цикла
' Вывод результатов – значений YMAX, XM
PRINT USING "  YMAX = ##.####^^^^ "; YMAX;
PRINT USING "  при X = #.## "; XM
END
```

Результаты работы программы представлены в п.5 (стр.19).

Лабораторная работа 5

Исследование функции одной переменной, зависящей от параметра

Тема. Табулирование функции одной переменной

Индивидуальные задания – табл. 3.5 (стр.102)
Математические функции – табл. 5.1 (стр.136)

1. Задание

Составить программу решения следующей задачи.

Вычислить таблицу значений функции $y = f(X, \alpha)$ и аргумента X на интервале $[X_{min}, X_{max}]$ с шагом ΔX при условии, что параметр α принимает ряд фиксированных значений α_1 и α_2. Таким образом, вычисление функции можно представить в виде:

$$y = \begin{cases} f(X, \alpha_1), & \text{если} \quad X_{\min} \le X \le X_{\text{ср}} \\ f(X, \alpha_2), & \text{если} \quad X_{\text{ср}} < X \le X_{\max}, \end{cases}$$

где $X_{ср}$ – значение аргумента X из интервала $X_{min} < X_{ср} < X_{max}$.

Протабулировать функцию **$f(X, \alpha) = \alpha \cdot X - \cos^2 \alpha X$** и найти количество её отрицательных значений на интервале **$[X_{min}, X_{max}]$**.

Исходные данные

Интервал		Шаг	«Среднее» значение аргумента X	Значения параметра α	
X_{min}	X_{max}	ΔX	$X_{ср}$	α_1	α_2
0	**1**	**0,2**	**0,4**	**0,5**	**1**

2. Решение

Обозначим: **K** – количество отрицательных значений функции.

Обозначения переменных				
Имя переменной		Тип переменной		Пояснения
в задаче	в программе	**Fortran**	**C**	
X_{min}, X_{max}	XMIN, XMAX	**REAL**	**DOUBLE**	Начальное и конечное значения аргумента
$X_{ср}$	XCP	**REAL**	**DOUBLE**	«Среднее» значение аргумента
ΔX	DX	**REAL**	**DOUBLE**	Шаг изменения аргумента
α_1, α_2	A1,A2	**REAL**	**DOUBLE**	Параметры
α	A	**REAL**	**DOUBLE**	Текущее значение параметра
X, Y	X, Y	**REAL**	**DOUBLE**	Текущие значения аргумента и функции
K	K	**INTEGER**	**INT**	Количество отрицательных значений функции

Количество вычисляемых значений (X,Y) [строк в таблице] определяется как:

$$\frac{\text{Xmax - Xmin}}{\Delta X} + 1 = 6$$

3. Ручной расчёт результатов

Вычислим значение y = f(X, α) в начальной и конечной точках.

Контрольные точки			
Аргумент X		Функция y = f(X, α)	
X_{min}	0	f (X_{min}, α_1)	–1
X_{max}	1	f (X_{max}, α_2)	0,708

Таблица значений аргумента X и функции y на интервале

[0;0,4]		[0,6;1]	
X	f(X, α_1)	X	f(X, α_2)
0	–1	0,6	–0,0812
0,2	–0,89	0,8	0,3146
0,4	–0,7605	1	0,7081

Очевидно, что количество отрицательных значений функции K = 4.

4. Алгоритм

При решении используется алгоритм вычисления количества. В п.4 – п.6 осуществляется выбор параметра α в зависимости от текущего значения аргумента X (проверка условия $X \leq X_{ср}$).

Алгоритм состоит из следующих действий:

1. Ввод данных – значений X_{min} , X_{max}, $X_{ср}$, ΔX, α_1, α_2.
2. Задание начального значения переменным $X = X_{min}$, K = 0.
3. *Начало цикла по X.*
4. Проверка условия $X \leq X_{ср}$. Если условие выполнено, перейти к п.5, иначе – к п.6.
5. Параметру α присваивается значение α_1: $\alpha = \alpha_1$. Перейти к п.7.
6. Параметру α присваивается значение α_2: $\alpha = \alpha_2$.
7. Вычисление значения функции y = f(x, α).
8. Вывод текущих значений аргумента X и функции Y.
9. Проверка условия Y < 0. Если условие выполнено, перейти к п.10, иначе – к п.11.
10. Увеличение текущего значения счётчика: K = K + 1.
11. Увеличение текущего значения аргумента: X = X + ΔX.
12. *Конец цикла по X.*
13. Вывод результата – количества K.
14. Конец.

Схема алгоритма представлена на рис. 2.8 (стр.27).

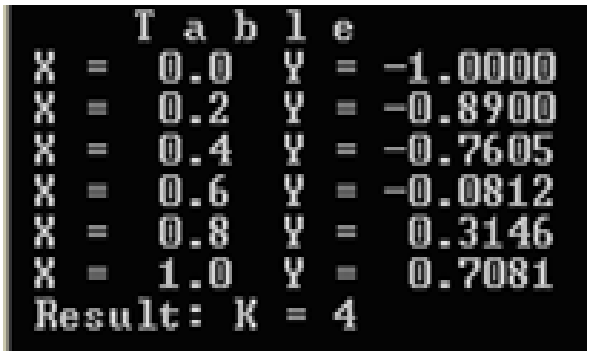
5. Листинг С-программы

Внимание. *При вводе программы следует иметь в виду, что С различает регистр букв, т.е., «**x**» и «**X**» – это две разные переменные.*

```
// Табулирование функции, зависящей от параметра
#include<stdio.h> // Файлы заголовков
#include<conio.h>
#include<math.h>
   void main()
{
// Объявление типа и задание данных
 int k = 0;
 double a, a1 = 0.5,a2 = 1,xcp = 0.4;
 double x,y,xmin = 0,xmax = 1,dx = 0.2;
 printf("     T a b l e \n"); // Вывод заголовка
 x = xmin;
 while(x<=xmax + dx/3) // Цикл с предусловием
   {
// Выбор параметра α
    if(x <= xcp) // Если X ≤ Xcp
       a = a1;
    else // Иначе,если X > Xcp
       a = a2;
    y = a*x - pow(cos(a*x),2);
// Вывод значений X и Y
    printf(" X =  %.1lf  Y = %7.4lf \n",x,y);
    if(y < 0)k++; // Вычисление значения K
    x += dx;
   }
 printf(" Result: K = %d ",k); // Вывод значения K
 getch();
 return;
}
```

Примечание. При выполнении оператора **while (x<=xmax)** в результате накопления ошибки может произойти «потеря» значения функции при $\mathbf{X = X_{max}}$. Поэтому значение **xmax** следует увеличить: **while (x<=xmax+dx/3).**

6. Макет печати (по С-программе)



Как видно, результаты ручного и машинного расчётов совпадают.

7. Листинг Fortran-программы

Программный код

```
! Табулирование функции, зависящей от параметра
PROGRAM PARAMETR
! Объявление типа и задание данных
INTEGER :: K  = 0
REAL :: A,A1 = 0.5,A2 = 1,XCP = 0.4
REAL :: X,Y,XMIN = 0,XMAX = 1,DX = 0.2
X = XMIN
WRITE(*,*)'        ТАБЛИЦА'   ! Вывод заголовка
DO WHILE (X <= XMAX + DX/3)      ! Начало цикла
! Выбор параметра A
   IF (X <= XCP) THEN ! Если X ≤ XCP
      A = A1
   ELSE                ! Иначе, если X > XCP
      A = A2
   END IF
   Y = A*X - COS(A*X)**2 ! Вычисление значения Y
! Вывод значений X,Y
   WRITE(*,2) X,Y
2 FORMAT (1X,' X = ',F3.1,2X,' Y = ',F7.4)
   IF(Y < 0) K = K + 1 ! Вычисление K
   X = X + DX
END DO                              ! Конец цикла
WRITE(*,"( A,I2)")'Результат K = ',K ! Вывод K
END
```

Результаты представлены в макете печати в п.6.

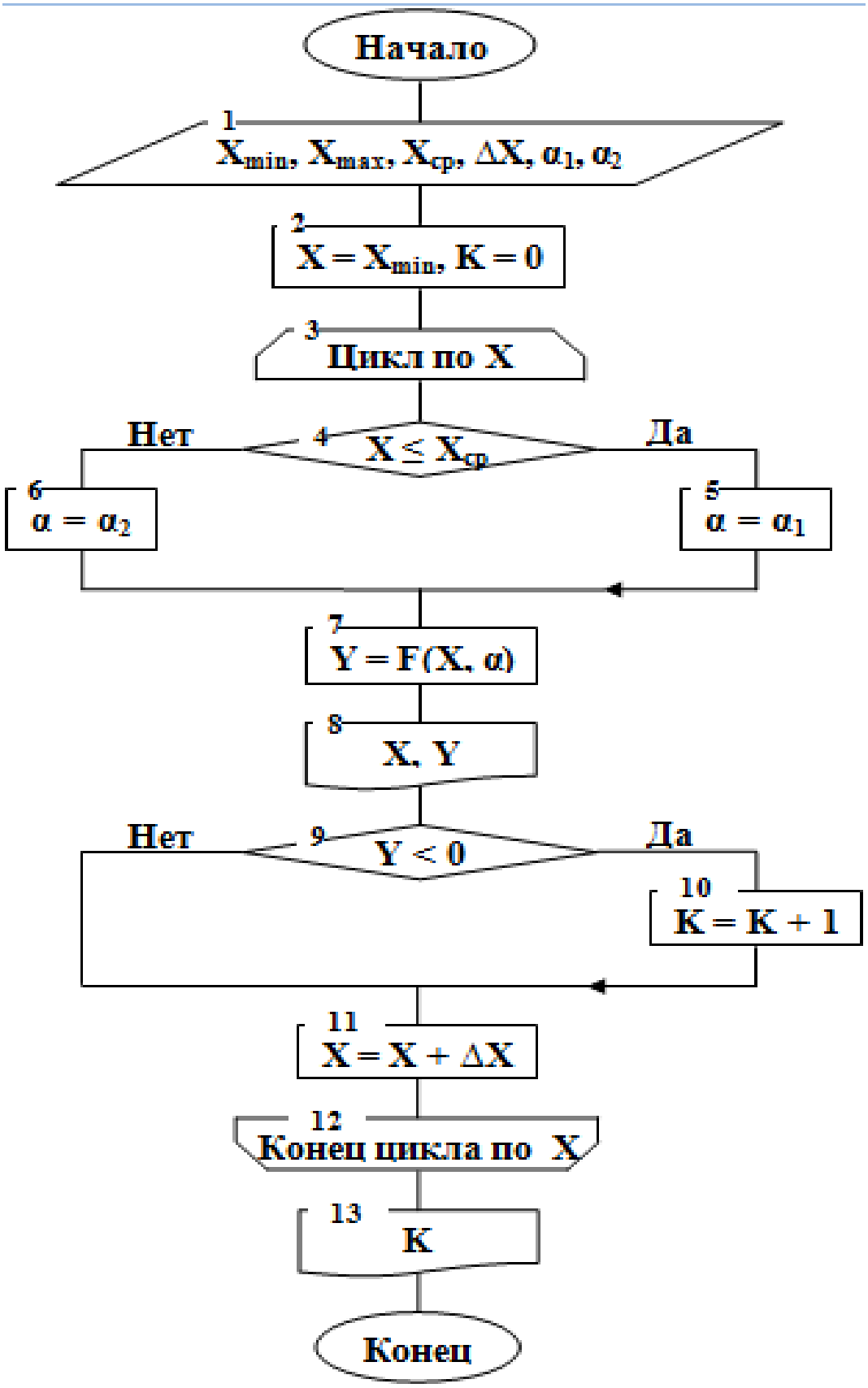


Рис. 2.8 Схема алгоритма к лабораторной работе 5 «Исследование функции одной переменной, зависящей от параметра»

Лабораторная работа 6

Исследование функции одной переменной с различным шагом варьирования аргумента

Тема. Табулирование функции одной переменной

Индивидуальные задания – табл. 3.6 (стр.105)

Математические функции – табл. 5.1 (стр.136)

1. Задание

Составить программу решения следующей задачи.

Вычислить таблицу значений функции Y = F(X) и аргумента X на интервале $[X_{min}, X_{max}]$ с шагом ΔX, причём шаг ΔX принимает ряд фиксированных значений ΔX_1 и ΔX_2 в зависимости от текущего значения аргумента X, а именно:

$$\Delta X = \begin{cases} \Delta X1, & если\ Xmin \leq X \leq Xcp \\ \Delta X2, & если\ Xcp < X \leq Xmax \end{cases}$$

где $X_{ср}$ – одно из текущих значений аргумента X на интервале $X_{min} < X_{ср} < X_{max}$.

Протабулировать функцию F(x) = shx – $\sqrt{x}$ и найти сумму её значений на интервале $[X_{min}; X_{max}]$. Определить знак суммы и вывести текстовое сообщение.

Исходные данные

Интервал		«Среднее» значение аргумента	Шаг изменения аргумента	
X_{min}	X_{max}	$X_{ср}$	ΔX_1	ΔX_2
0,5	**1,5**	**1,1**	**0,2**	**0,1**

2. Решение

Обозначения переменных				
Имя переменной		Тип переменной		Пояснение
в задаче	в программе	**Fortran**	**C**	
$\mathbf{X_{min}}$, $\mathbf{X_{max}}$	XMIN, XMAX	**REAL**	**DOUBLE**	Начальное и конечное значение X
$\mathbf{X_{cp}}$	XCR	**REAL**	**DOUBLE**	Значение аргумента на интервале $X_{min} < X_{cp} < X_{max}$
$\Delta\mathbf{X_1}$	DX1	**REAL**	**DOUBLE**	Шаг аргумента на интервале $[X_{min}; X_{cp}]$
$\Delta\mathbf{X_2}$	DX2	**REAL**	**DOUBLE**	Шаг аргумента на интервале $[X_{cp} + \Delta X_2; X_{max}]$
ΔX	DX	**REAL**	**DOUBLE**	Текущее значение шага
X, Y	X, Y	**REAL**	**DOUBLE**	Значения аргумента и функции
S	S	**REAL**	**DOUBLE**	Сумма значений функции

3. Ручной расчёт результатов

Вычислим значение Y = F(X) в начальной и конечной точках.

Контрольные точки	Аргумент	Функция
X_{min}	0,5	−0,186
X_{max}	1,5	0,905

Результаты можно представить в табличном и в графическом виде.

Графический способ

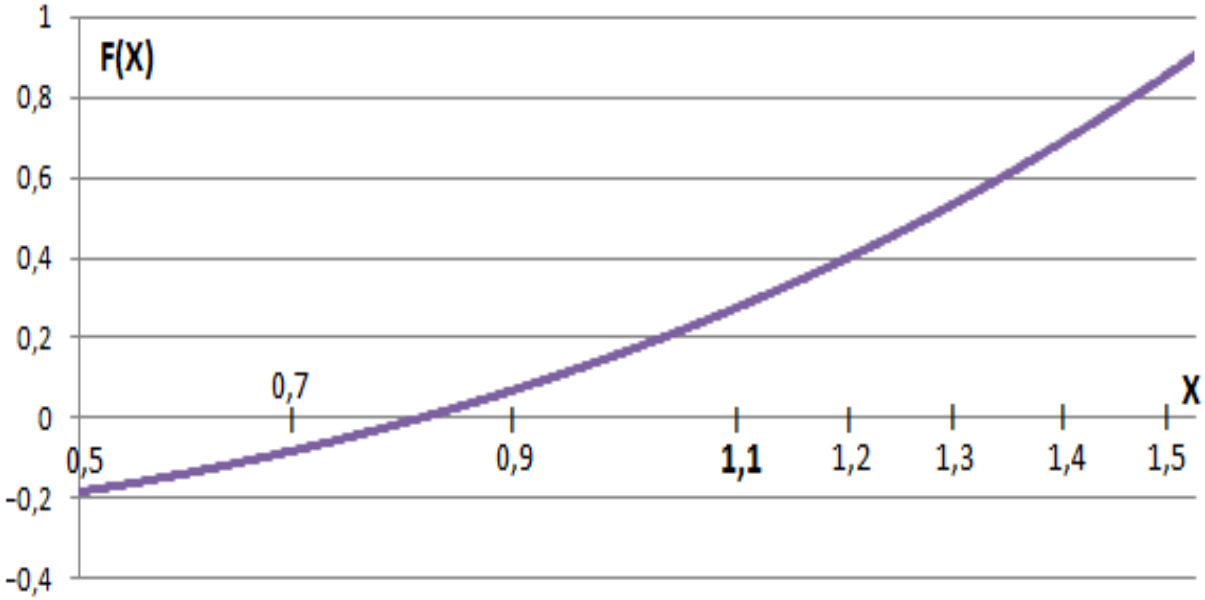


Таблица значений функции Y = F(X) на интервале

[0,5;1,1] с шагом $\Delta X_1 = 0,2$		**[1,2;1,5] с шагом $\Delta X_2 = 0,1$**	
X	Y	X	Y
0,5	–0,186	1,2	0,414
0,7	–0,078	1,3	0,558
0,9	0,078	1,4	0,721
1,1	**0,287**	1,5	0,905

Вычислим сумму значений функции: S = 2,698, S > 0. При решении задачи используется алгоритм вычисления суммы.

4. Алгоритм

В п.8 – п.9 выполняется смена шага $\Delta X = \Delta X_2$, если текущее значение аргумента X равно X_{cp} ($X = X_{cp}$). В п.12 – п.14 определяется знак суммы S и выводится текстовое сообщение.

Алгоритм состоит из следующих действий:

1. Ввод данных – значений переменных X_{min} , X_{max}, X_{cp}, ΔX_1, ΔX_2.
2. Задание начального значения переменным: $X = X_{min}$; S = 0.
3. Задание начального значения шага $\Delta X = \Delta X_1$.
4. *Начало цикла по X.*
5. Вычисление значения функции Y = F(X).
6. Вычисление суммы S = S + Y.
7. Вывод текущего значения аргумента X и функции Y.

8. Проверка условия $X = X_{ср}$. Если условие выполняется, то перейти к п.9, иначе – к п.10.
9. Изменение текущего значения шага $\Delta X = \Delta X_2$.
10. Увеличение аргумента X на шаг $X = X + \Delta X$.
11. *Конец цикла по X.*
12. Проверка условия $S > 0$. Если условие выполняется, то перейти к п.13, иначе – к п.14.
13. Вывод сообщения $S > 0$ (сумма положительна), перейти к п.15.
14. Вывод сообщения $S < 0$ (сумма отрицательна).
15. Конец.

Схема алгоритма приведена на рис. 2.9.

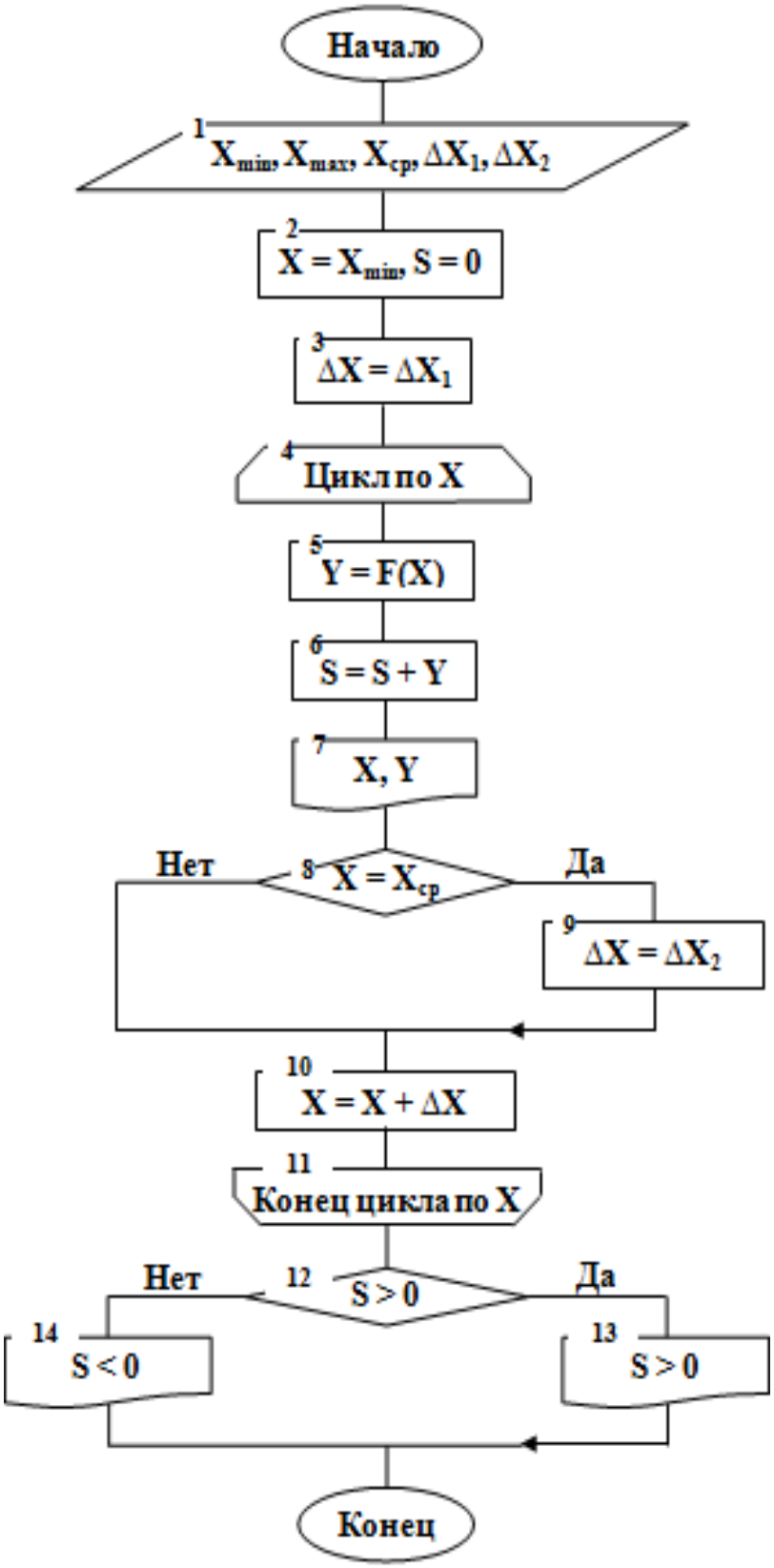


Рис. 2.9 Схема алгоритма к лабораторной работе 6 «Исследование функции одной переменной с различным шагом варьирования аргумента»

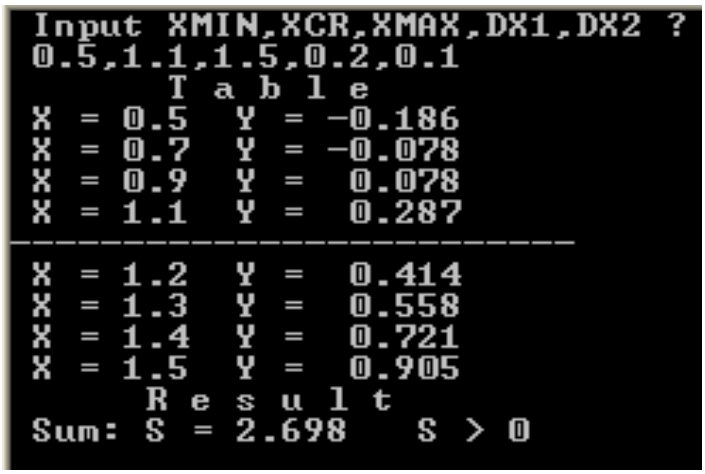
5. Листинг С-программы

Внимание. *При вводе программы следует иметь в виду, что С различает регистр букв, т.е., «**x**» и «**X**» – это две разные переменные.*

```
// Табулирование функции с различным шагом
#include<stdio.h> // Файлы заголовков
#include<conio.h>
#include<math.h>
void main()
{
 double s = 0,xmin,xcr,xmax; // Объявление данных
 double dx,dx1,dx2,x,y;
// Ввод данных с клавиатуры по запросу с экрана,
// числа при вводе разделяются запятой
printf(" Input XMIN,XCR,XMAX,DX1,DX2 ?\n ");
scanf("%f,%f,%f,%f,%f",&xmin,&xcr,&xmax,&dx1,&dx2);
x = xmin;
 dx = dx1;
 do // Цикл с постусловием
  {
   y = sinh(x)-sqrt(x);
   s + = y; // Вычисление суммы S
   printf(" X = %.1lf  Y = %6.3lf \n",x,y);
   if(abs(x - xcr)<=0.001) // Если X = Xcp
     {
      printf("-------------------- \n");
      dx = dx2;
     }
   x = x + dx; // Увеличение аргумента на шаг
  }
 while(x<=xmax + dx/3); //  Конец цикла
 printf("     R e s u l t  \n ");
// Определение знака суммы S
 if(s > 0) // Если S > 0
    printf(" Sum: S =  %.3lf   S > 0 ",s);
 else // Иначе,если S < 0
    printf(" Sum: S =  %.3lf   S < 0 ",s);
getch();
return;
}
```

Примечание. При выполнении оператора **while (x<=xmax)** в результате накопления ошибки может произойти «потеря» значения функции при $\mathbf{X = X_{max}}$. Поэтому значение **xmax** следует увеличить, например: **while (x<=xmax + dx/3).**

6. Макет печати (по С-программе)



Как видно, результаты ручного и машинного расчётов совпадают.

Контрольное задание для проверки знаний

Расчёт ординат эпюры давления грунта на подпорные стенки

Тема. Табулирование функции одной переменной

В инженерной практике процесс табулирования функции, т.е. вычисление значения функции на заданном интервале с заданным шагом, широко используется при создании вычислительных программ.

Составить схему алгоритма и программу для следующей задачи.

Для исследования подпорных стенок на устойчивость и прочность возникает необходимость определения различных нагрузок, которые воспринимают проектируемые сооружения. Одна из основных нагрузок – давление грунта, расположенного за подпорной стенкой.

Задание

Рассчитать ординаты эпюры давления грунта на подпорные стенки.

Исходные данные

- $\gamma = 19$ – удельный вес грунта для песка,
- $\varphi = 10^{\circ}$ – угол внутреннего трения грунта для песка,
- **h** – толщина слоя грунта.

Ординаты эпюры определяются по формуле:

$$\sigma(h) = \gamma \cdot h \cdot tg\left(45^{\circ} - \frac{\varphi}{2}\right)^2$$

Математическая формулировка задачи:

Вычислить значения аргумента **h** и функции **σ(h)** при условии, что аргумент **h** изменяется в интервале $0 \le h \le 4,5$ с шагом $\Delta\mathbf{h} = 0,5$ для функции вида:

$$\sigma(h) = \gamma \cdot h \cdot tg\left(45^\circ - \frac{\varphi}{2}\right)^2$$

Таблица значений аргумента h и функции σ(h)

h	0	0,5	1	1,5	2	2,5	3	3,5	4	4,5
σ(h)	0	6,7	13,4	20,1	26,8	33,4	40,1	46,8	53,5	60,2

Алгоритм циклической структуры, параметр цикла – аргумент h.

Лабораторная работа 7

Исследование функции двух переменных

Тема. Табулирование функции двух переменных

Индивидуальные задания – табл. 3.7 (стр.107)

Математические функции – табл. 5.1 (стр.136)

1. Задание

Составить программу решения следующей задачи:

Вычислить и напечатать таблицу значений функции Z = f(X,Y) и аргументов X,Y в заданной области при условии, что:

- аргумент X изменяется в интервале $X_0 \le X \le X_k$ с шагом ΔX;
- аргумент Y изменяется в интервале $Y_0 \le Y \le Y_k$ с шагом ΔY;

Протабулировать функцию вида $Z = X^2 + Y^2$ и найти её среднее арифметическое в заданной области.

Для выполнения лабораторной работы необходимо:

1. Представить заданную область на плоскости (X, Y).
2. Пронумеровать точки в области в соответствии с алгоритмом (рис. 2.11).
3. Представить макет печати таблицы значений аргументов.

Порядок расположения координат точек (X,Y) в таблице должен соответствовать их нумерации в заданной области.

Исходные данные

Интервал		**Шаг**	**Интервал**		**Шаг**
$\mathbf{X_0}$	$\mathbf{X_k}$	$\Delta\mathbf{X}$	$\mathbf{Y_0}$	$\mathbf{Y_k}$	$\Delta\mathbf{Y}$
1	2	0,5	1	3	1

2. Решение

Обозначим: **S, K, SR** – сумма, количество значений функции **Z** в области и её среднее арифметическое; **SR = S/K**.

Обозначения переменных				
Имя переменной		Тип переменной		Пояснения
в схеме	в программе	**Fortran**	**C**	
X_0, X_k	X0, XK	**REAL**	**FLOAT**	Начальное и конечное значение X
Y_0,Y_k	Y0, YK	**REAL**	**FLOAT**	Начальное и конечное значение Y
ΔX, ΔY	DX, DY	**REAL**	**FLOAT**	Шаг изменения аргументов X и Y
X, Y, Z	X, Y, Z	**REAL**	**FLOAT**	Значения аргументов и функции
K	K	**INTEGER**	**INT**	Количество значений функции
S, SR	S, SR	**REAL**	**FLOAT**	Сумма значений функции и её среднее арифметическое в области

График функции $Z = X^2 + Y^2$ – поверхность (рис. 2.10).

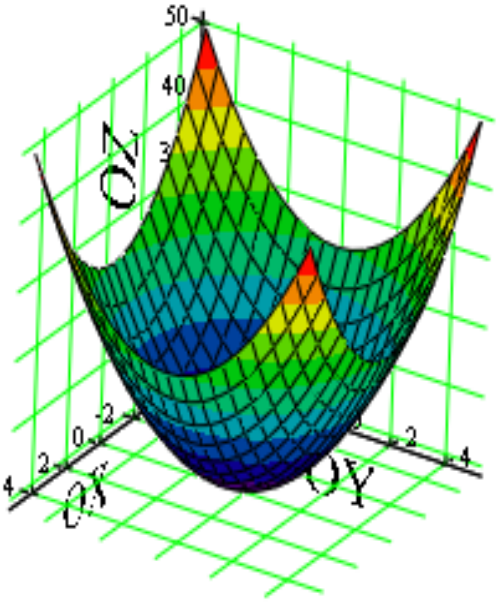


Рис. 2.10 График функции

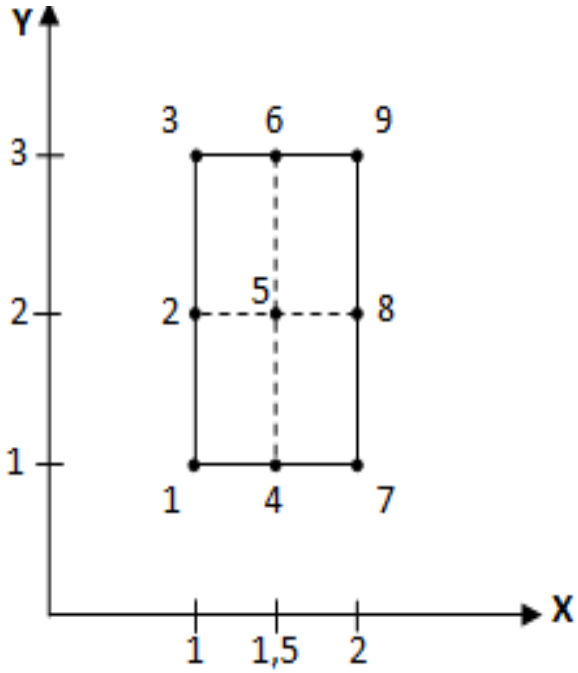


Рис. 2.11 Нумерация точек (значение X фиксировано)

Количество точек (X, Y), для которых нужно вычислить значение функции Z, определяется как:

$$\left(\frac{X_k - X_0}{\Delta X} + 1\right) \cdot \left(\frac{Y_k - Y_0}{\Delta Y} + 1\right) = 9$$

Координаты точек в области приведены в следующей таблице:

Для фиксированного значения аргумента X									
Координаты точек									
Номер точки	1	2	3	4	5	6	7	8	9
Значение X	**X = 1**			**X = 1,5**			**X = 2**		
Координаты	(1;1)	(1;2)	(1;3)	(1,5;1)	(1,5;2)	(1,5;3)	(2;1)	(2;2)	(2;3)

3. Алгоритм

Для решения задачи используется алгоритм вычисления суммы и количества.

Примечание. Для вычисления среднего арифметического функции во всей заданной области направление «обхода» точек на плоскости значения не имеет.

В алгоритме выбран вариант, когда фиксировано значение аргумента X (рис. 2.11). Приведём словесное описание алгоритма. Алгоритм содержит двойной цикл вложенной структуры. Параметр внешнего цикла – аргумент X, а внутреннего – аргумент Y.

Алгоритм состоит из следующих действий:

1. Ввод данных – значений переменных X_0, X_k, ΔX, Y_0, Y_k, ΔY;
2. Задание начального значения переменным S = 0, K = 0.
3. *Начало внешнего цикла по X.*
4. *Начало внутреннего цикла по Y.*
5. Вычисление значения функции Z = f(X,Y).
6. Вывод текущих значений аргументов X,Y и функции Z.
7. Вычисление суммы и количества: S = S + Z, K = K + 1.
8. *Конец внутреннего цикла по Y.*
9. *Конец внешнего цикла по X.*
10. Вывод результата – значения SR = S/K.
11. Конец.

Схема алгоритма представлена на рис. 2.12 (стр.38).

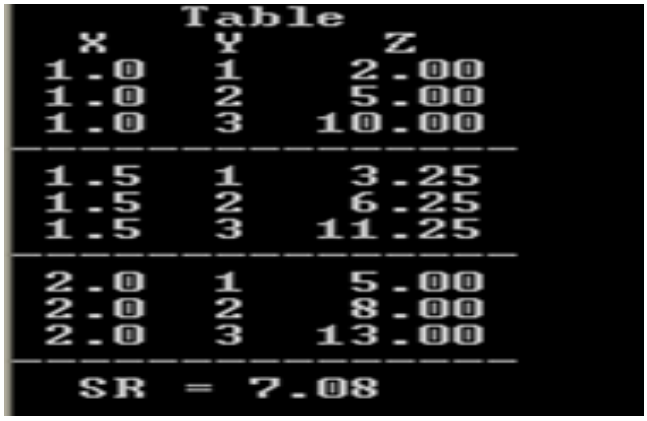
4. Листинг С-программы

Внимание. *При вводе программы следует иметь в виду, что С различает регистр букв, т.е., «**x**» и «**X**» – это две разные переменные.*

Программный код

```
# include<stdio.h> // Файлы заголовков
# include<conio.h>
# include<math.h>
  void main()
{
int k = 0; // Объявление типа и задание данных
float x0 = 1,xk = 2,dx = 0.5;
float y0 = 1,yk = 3,dy = 1;
float x, y, z, s = 0;
printf("     Table \n"); // Вывод заголовка
printf("  X    Y     Z \n");
for(x = x0; x<=xk; x += dx) // Внешний цикл по X
   {
 for(y = y0; y<=yk; y += dy) // Внутренний цикл по Y
      {
       z = x*x + pow(y,2); // Вычисление Z, S, K
// Вывод значений X, Y, Z
       printf(" %.1f  %.0f  %5.2f \n", x, y, z);
       s += z; k++;
      }
printf("---------------\n"); // Вывод строки "-"
   }
printf("  SR = %.2f ", s/k); // Вывод SR
getch();
return;
}
```

5. Макет печати результатов



Как видно, расположение точек в макете соответствует их нумерации в заданной области (рис. 2.11).

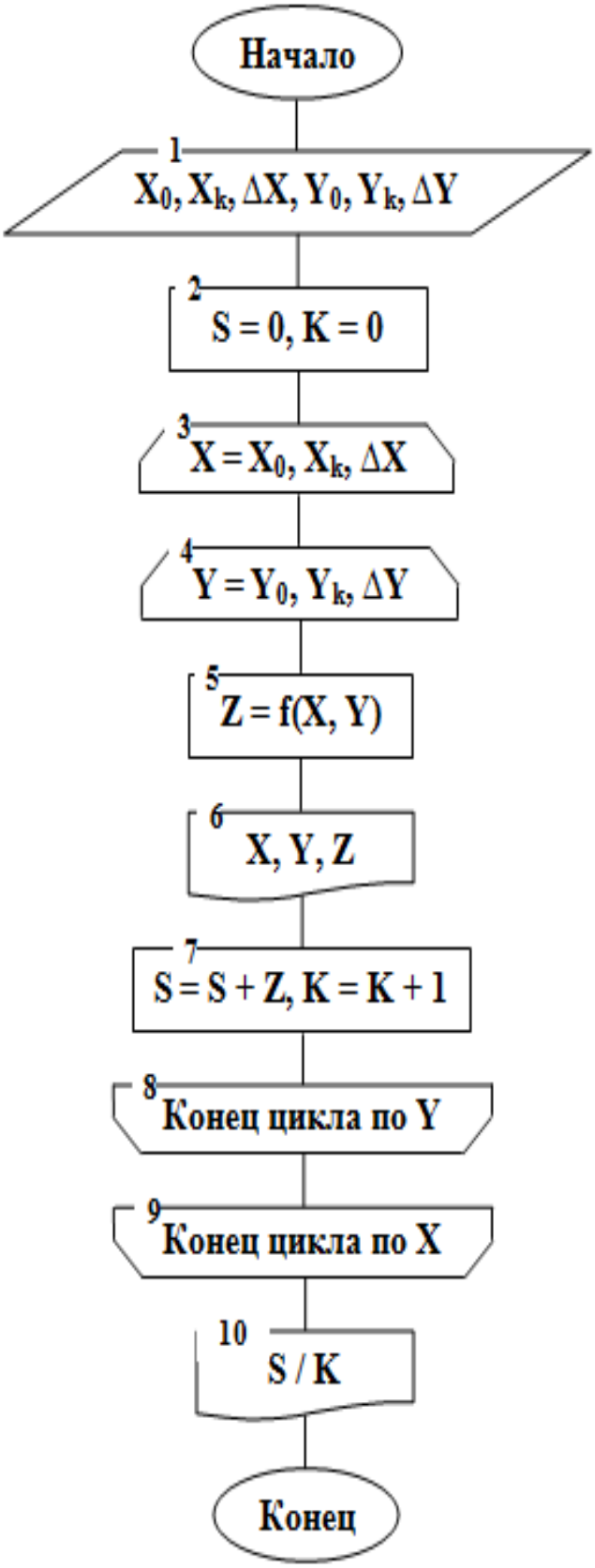


Рис. 2.12 Схема алгоритма к лабораторной работе 7 «Исследование функции двух переменных»

Лабораторная работа 8

Исследование функции двух переменных при фиксированном значении аргумента

Тема. Табулирование функции двух переменных

Индивидуальные задания – табл. 3.8 (стр.109)

Математические функции – табл. 5.1 (стр.136)

1. Задание

Составить программу решения следующей задачи:

Вычислить и напечатать таблицу значений функции Z = f(X,Y) и аргументов X,Y в заданной области при условии, что:

• аргумент X изменяется в интервале $X_0 \le X \le X_k$ с шагом ΔX;

• аргумент Y изменяется в интервале $Y_0 \le Y \le Y_k$ с шагом ΔY;

Рассмотрим два варианта задания.

Протабулировать функцию $Z = X^2 + Y^2$ и найти её среднее арифметическое:

• для каждого фиксированного значения аргумента X;

• для каждого фиксированного значения аргумента Y.

Для выполнения лабораторной работы необходимо:

1. Представить графически заданную область на плоскости.
2. Пронумеровать точки в области в соответствии с алгоритмом (рис. 2.13-2.14).
3. Представить макет печати таблицы значений аргументов. Расположение точек (X,Y) в таблице должно соответствовать их нумерации в заданной области.

Исходные данные

Интервал		Шаг	Интервал		Шаг
X_0	X_k	ΔX	Y_0	Y_k	ΔY
1	2	0,5	1	3	1

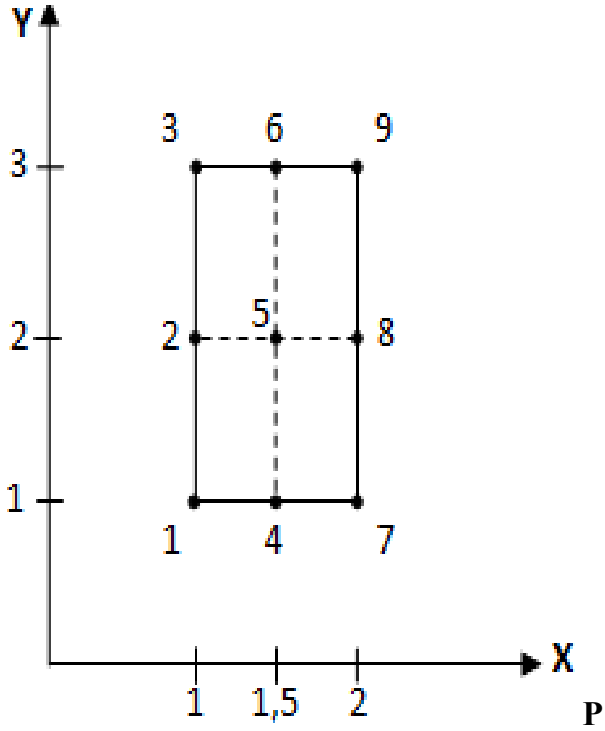
2. Решение

Обозначим: **S, K, SR** – сумма, количество значений функции **Z** и её среднее арифметическое для фиксированного значения аргумента (X или Y); **SR = S/K**.

Обозначения переменных				
Имя переменной		Тип переменной		Пояснения
в схеме	в программе	**Fortran**	**C**	
X_0, X_k	X0, XK	**REAL**	**FLOAT**	Начальное и конечное значение X
Y_0,Y_k	Y0, YK	**REAL**	**FLOAT**	Начальное и конечное значение Y
ΔX, ΔY	DX, DY	**REAL**	**FLOAT**	Шаг изменения X и Y
X, Y, Z	X, Y, Z	**REAL**	**FLOAT**	Значения аргументов и функции
K	K	**INTEGER**	**INT**	Количество значений функции для фиксированного аргумента X или Y
S, SR	S, SR	**REAL**	**FLOAT**	Сумма значений функции и её среднее арифметическое для фиксированного аргумента (X или Y)

График функции $Z = X^2 + Y^2$ – поверхность (рис. 2.10).

Графическое представление заданной области



ис. 2.13 Схема нумерации точек для фиксированного значения X

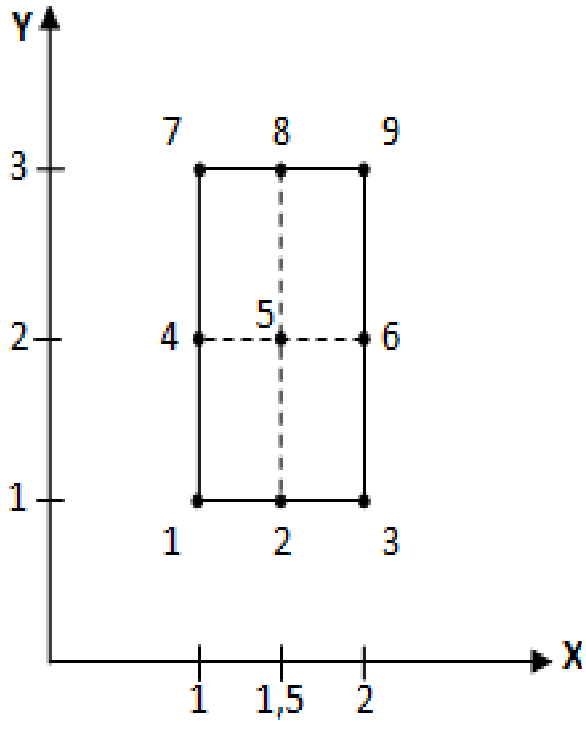


Рис. 2.14 Схема нумерации точек для фиксированного значения Y

Количество точек (X, Y) в заданной области определяется как:

$$(\frac{X_k - X_0}{\Delta X} + 1) \cdot (\frac{Y_k - Y_0}{\Delta Y} + 1) = 9$$

3. Алгоритм

Для фиксированного значения аргумента X параметр внешнего цикла – X, а внутреннего – Y. **Для фиксированного значения аргумента Y** внешний цикл организуется по параметру Y, а внутренний – по X.

Схемы алгоритма представлены на рис. 2.15 - 2.16:

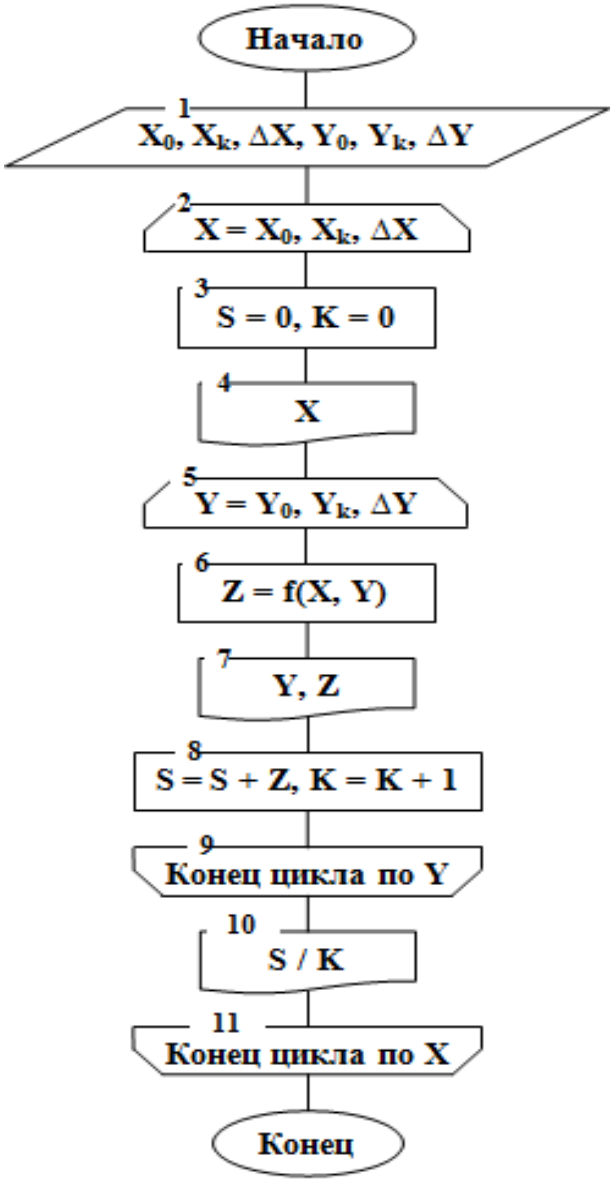


Рис. 2.15 Схема алгоритма к лабораторной работе 8 «Исследование функции двух переменных» при фиксированном X

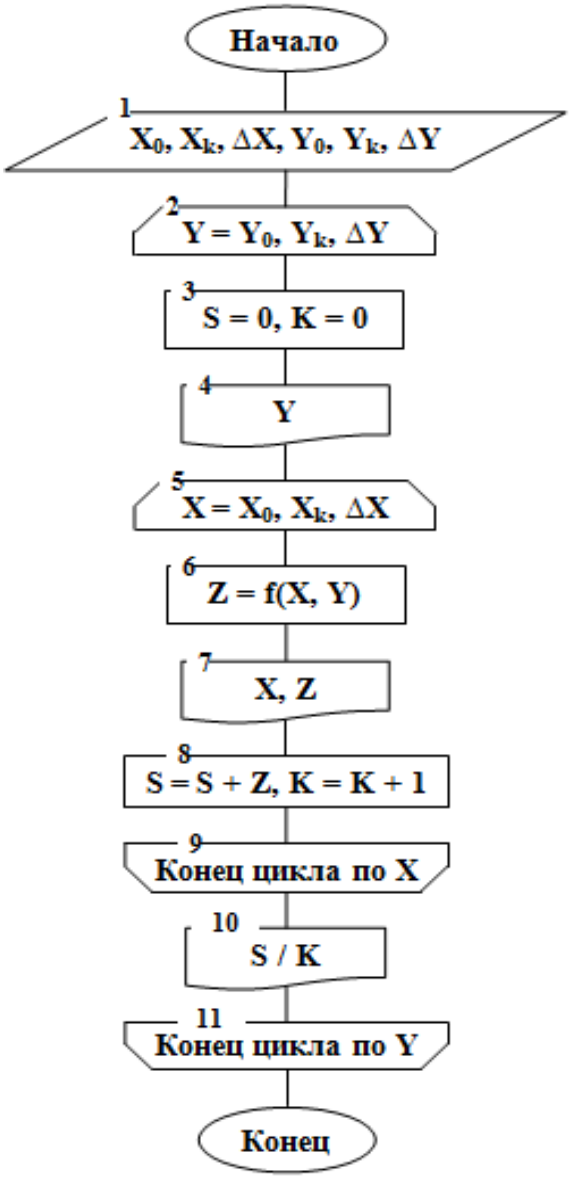


Рис. 2.16 Схема алгоритма к лабораторной работе 8 «Исследование функции двух переменных» при фиксированном Y

4. Листинг Basic-программы

Для фиксированного значения X	Для фиксированного значения Y
Программный код	
' Ввод данных **READ** X0,XK,DX,Y0,YK,DY **DATA** 1,2,0.5,1,3,1 ' Внешний цикл по X **FOR** X = X0 **TO** XK **STEP** DX S = 0: K = 0 **PRINT TAB**(6); " X = ";X ' Внутренний цикл по Y **FOR** Y = Y0 **TO** YK **STEP** DY Z = X ^ 2 + Y ^ 2 **PRINT** "Y =";Y," Z =";Z S = S + Z: K = K + 1 **NEXT** Y 'Конец цикла по Y **PRINT TAB**(6);"SR = "; **PRINT USING** "##.##";S/K **NEXT** X ' Конец цикла по X **END**	' Ввод данных **READ** X0,XK,DX,Y0,YK,DY **DATA** 1,2,0.5,1,3,1 ' Внешний цикл по Y **FOR** Y = Y0 **TO** YK **STEP** DY S = 0: K = 0 **PRINT TAB**(6); " Y = ";Y ' Внутренний цикл по X **FOR** X = X0 **TO** XK **STEP** DX Z = X ^ 2 + Y ^ 2 **PRINT** "X =";X," Z =";Z S = S + Z: K = K + 1 **NEXT** X 'Конец цикла по X **PRINT TAB**(6);"SR = "; **PRINT USING** "##.##";S/K **NEXT** Y ' Конец цикла по Y **END**

5. Макеты печати (по Basic-программе)

Для фиксированного X	Для фиксированного Y
X = 1	**Y = 1**
Y = 1 Z = 2	X = 1 Z = 2
Y = 2 Z = 5	X = 1.5 Z = 3.25
Y = 3 Z = 10	X = 2 Z = 5
SR = 5.67	**SR = 3.42**
X = 1.5	**Y = 2**
Y = 1 Z = 3.25	X = 1 Z = 5
Y = 2 Z = 6.25	X = 1.5 Z = 6.25
Y = 3 Z = 11.25	X = 2 Z = 8
SR = 6.92	**SR = 6.42**
X = 2	**Y = 3**
Y = 1 Z = 5	X = 1 Z = 10
Y = 2 Z = 8	X = 1.5 Z = 11.25
Y = 3 Z = 13	X = 2 Z = 13
SR = 8.67	**SR = 11.42**

Расположение точек в макете соответствует их нумерации в области (рис. 2.13-2.14, стр.40).

Лабораторная работа 9

Вычисление координат искомой точки, наиболее удалённой от начала координат

Тема. Одномерный массив: min, max

Индивидуальные задания – табл. 3.9 (стр.111)
Математические функции – табл. 5.1 (стр.136)

1. Задание

Составить программу решения следующей задачи.

На плоскости (x, y) заданы точки $\{M_1(X_1, Y_1), \ldots, M_n(X_n, Y_n)\}$ и круг радиуса **R** с центром в начале координат, ***n*** – количество точек. Координаты точек задаются в виде массивов абсцисс $\mathbf{(X_1, \ldots, X_n)}$ и ординат $\mathbf{(Y_1, \ldots, Y_n)}$, *в соответствии с условием задачи* и так, чтобы *искомая точка была одна*.

Среди точек правой полуплоскости, лежащих внутри круга, определить точку, наиболее удалённую от начала координат.

Для выполнения лабораторной работы необходимо:

- задать точки $\{M_1(X_1, Y_1), \ldots, M_n(X_n, Y_n)\}$ и радиус круга R.
- представить заданные точки и круг на плоскости (X,Y).
- найти и обозначить на графике искомую точку.

Исходные данные

Количество точек	Координаты точек $M_i(X_i, Y_i)$				
	Абсциссы				
N = 5	$\mathbf{X_1}$	$\mathbf{X_2}$	$\mathbf{X_3}$	$\mathbf{X_4}$	$\mathbf{X_5}$
	4	**1**	**–2**	**1**	**–2**
Радиус круга	Ординаты				
R = 3	$\mathbf{Y_1}$	$\mathbf{Y_2}$	$\mathbf{Y_3}$	$\mathbf{Y_4}$	$\mathbf{Y_5}$
	5	**2**	**1**	**–1**	**–3**

2. Решение

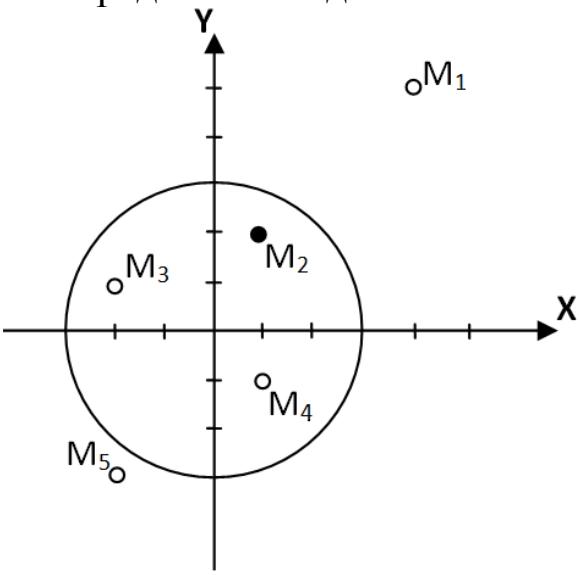
Обозначим: **RT** – расстояние от $M_i(X_i, Y_i)$ до начала координат.

K, RMAX – номер искомой точки, удалённой от начала координат на расстояние RMAX.

Обозначения переменных			
Имя переменной	Тип переменной		Пояснение
	Fortran	**C**	
N, i	**INTEGER**	**INT**	Количество точек и порядковый номер точки
R	**REAL**	**FLOAT**	Радиус круга
X_i, Y_i	**REAL**	**FLOAT**	Координаты точки с порядковым номером i
RT	**REAL**	**FLOAT**	Расстояние от $M_i(X_i, Y_i)$ до начала координат
RMAX	**REAL**	**FLOAT**	Наибольшее расстояние
K	**INTEGER**	**INT**	Порядковый номер искомой точки
X_k,Y_k	**REAL**	**FLOAT**	Координаты искомой точки

3. Ручной расчёт результатов

Представим заданные точки и круг на плоскости (X,Y).



Координаты точек заданы в соответствии с условием задачи. Это означает, что среди заданных две точки, $M_2(1; 2)$ и $M_4(1; -1)$, расположены в правой полуплоскости и внутри круга.

Расстояние от точки $M_i(X_i, Y_i)$ до начала координат вычисляется по формуле **RT** = $\sqrt{X_i^2 + Y_i^2}$.

Для точек правой полуплоскости выполняется условие $X_i > 0$.

Представим точки, из которых осуществляется выбор искомой:

Точки, лежащие в правой полуплоскости и внутри круга			
Номер точки i	Точка $M_i(X_i, Y_i)$	Расстояние до начала координат	
2	$M_2(1; 2)$	$\sqrt{5}$	2,24
4	$M_4(1;-1)$	$\sqrt{2}$	1,41

Искомой является точка $M_2(1; 2)$ с номером K = 2, удалённая от начала координат на расстояние RMAX = 2,24.

4. Алгоритм

При решении задачи используется алгоритм вычисления наибольшего значения. Для контроля и наглядности результатов в п.8 выводятся точки, из которых осуществляется выбор искомой.

Алгоритм состоит из следующих действий:

1. Ввод координат – значений X_i и Y_i (i = 1,…,n) и радиуса R.
2. Вывод координат точек X_i, Y_i (i = 1,…,n).
3. Задание начального значения переменной RMAX = −1.0E38 (заведомо малого числа).
4. *Начало цикла.*
5. Вычисление значения расстояния RT.
6. Проверка условия $X_i > 0$. Если условие выполнено, перейти к п.7, иначе – к п.11.
7. Проверка условия RT < R. Если условие выполнено, перейти к п.8, иначе – к п.11.
8. Вывод координат X_i, Y_i и расстояния RT.
9. Проверка условия RT > RMAX. Если условие выполнено, перейти к п.10, иначе – к п.11.
10. Переменным присваиваются значения RMAX = RT, K = i.
11. *Конец цикла.*
12. Вывод искомой точки $M_k(X_k, Y_k)$ и расстояния RMAX.
13. Конец.

Схема алгоритма представлена на рис. 2.17 (стр.47).

5. Макет печати результатов (по С-программе)

```
  X         Y
-------------
  4         5
  1         2
 -2         1
  1        -1
 -2        -3

          Point
X(2) =  1  Y(2) =  2  RT = 2.24
X(4) =  1  Y(4) = -1  RT = 1.41
          Result
X(2) =   1  Y(2) =   2
RMAX = 2.24
```

Результаты ручного и машинного расчётов совпадают. В этом легко убедиться: на графике искомая точка обозначена (стр.44).

6. Листинг С-программы «Вычисление координат точки»

Внимание. *При вводе программы следует иметь в виду, что С различает регистр букв, т.е., «**x**» и «**X**» – это две разные переменные. Нижняя граница индекса i (порядкового номера элемента в массиве) задаётся равной нулю.*

```
# include<stdio.h> // Заголовочные файлы
# include<conio.h>
# include<math.h>
  void main()
{
int i,k,n = 5; // Объявление типа и задание данных
 float rt,r = 3,rmax = -1.0e38;
 float x[5] = {4,1,-2,1,-2};
 float y[5] = {5,2,1,-1,-3};
 printf("  X \t   Y\n"); // Вывод заголовка
 printf("--------------\n");
 for(i = 0;i<n;i++) // Цикл вывода координат точек
   printf(" %3.0f \t %3.0f\n ",x[i],y[i]);
  printf(" \n      \t Point  \n");
 for(i = 0;i<n;i++)// Цикл по i (номеру точки)
   {
    rt = sqrt(pow(x[i],2) + pow(y[i],2));
    if(x[i]>0 && rt<r) // Если Xi > 0 и RT < R
    {
// Вывод точек правой полуплоскости внутри круга
     printf(" X(%d) = %2.0f ", i+1, x[i]);
     printf(" Y(%d) = %2.0f ", i+1, y[i]);
     printf(" RT = %4.2f \n ", rt);
     if(rt>rmax) // Если RT > RMAX
       {
        rmax = rt; k = i;
       }
    }
   }
// Вывод координат искомой точки (Xk, Yk)
 printf(" \t Result \n");
 printf(" X(%d) = %3.0f ", k+1, x[k]);
 printf(" Y(%d) = %3.0f ", k+1, y[k]);
  printf(" \n RMAX = %4.2f ",rmax);
 getch();
 return;
 }
```

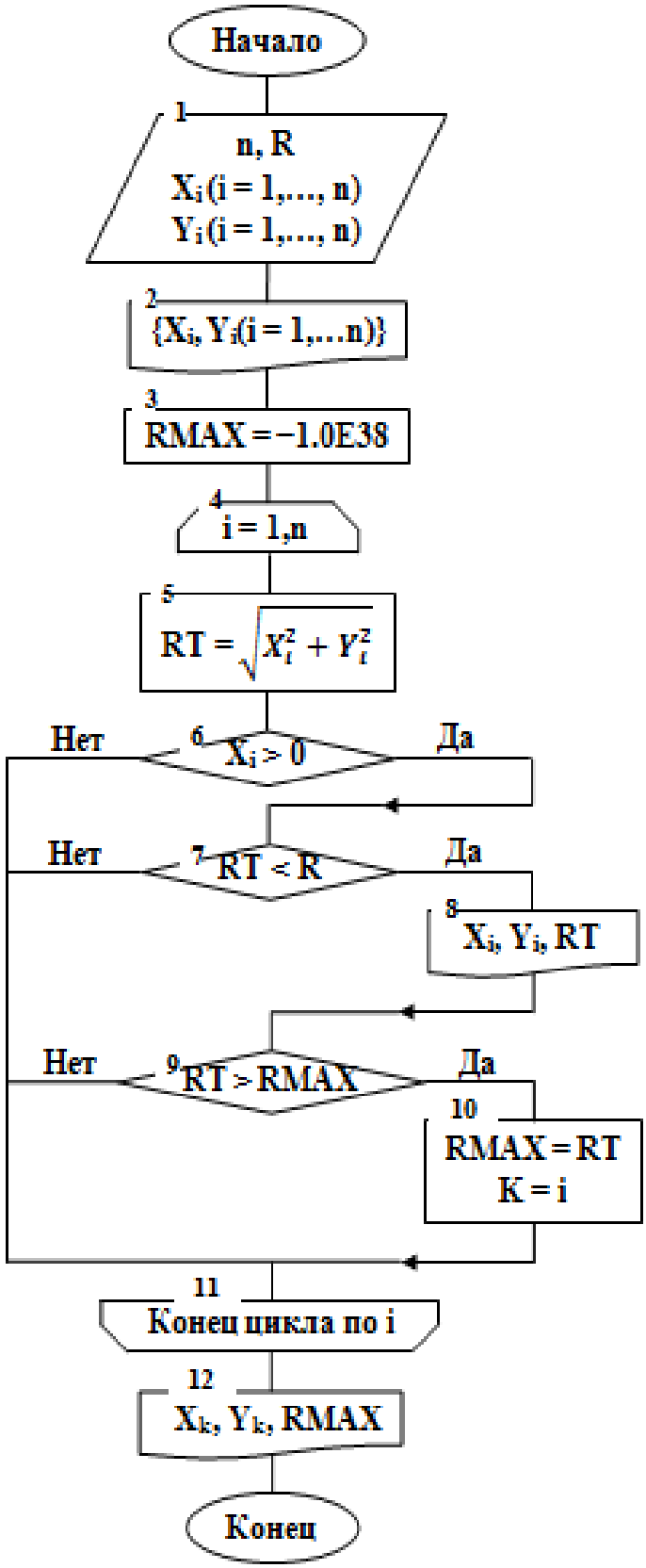


Рис. 2.17 Схема алгоритма к лабораторной работе 9 «Вычисление координат точки, наиболее удалённой от начала координат»

Лабораторная работа 10

Вычисление координат искомой точки (ближайшей к заданной)

Тема. Одномерный массив: min, max

Индивидуальные задания – табл. 3.10 (стр.113)

1. Задание

Составить программу решения следующей задачи.

На плоскости (x, y) заданы точки$\{M_1(X_1, Y_1), \ldots, M_n(X_n, Y_n)\}$ и точка $Q(X_q,Y_q)$. Координаты точек задаются в виде массивов абсцисс **(X_1, …, X_n)** и ординат **(Y_1, …, Y_n),** *в соответствии с условием задачи* и так, чтобы *искомая точка была одна*.

Среди точек первого и третьего квадранта определить точку, ближайшую к точке $Q(X_q,Y_q)$, количество точек n = 5.

Для выполнения лабораторной работы необходимо:

- задать все точки и представить их на плоскости (x,y).
- найти и обозначить на графике искомую точку.

Исходные данные

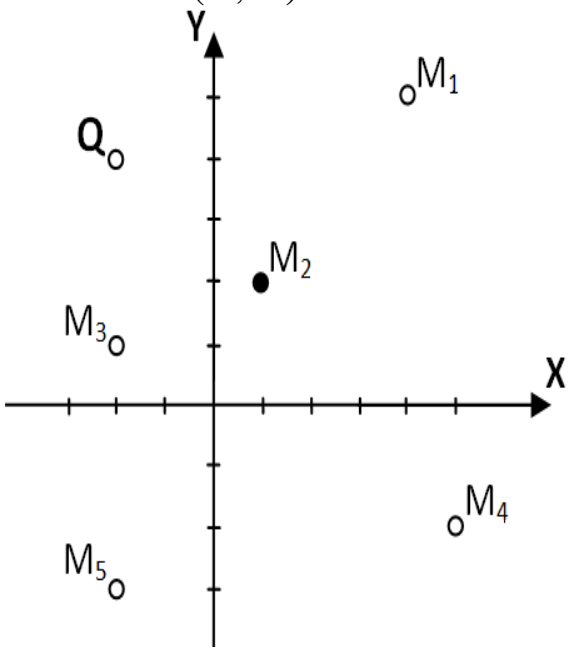
Точка Q	**Координаты точек $M_i(X_i, Y_i)$**				
X_q	**X_1**	**X_2**	**X_3**	**X_4**	**X_5**
–2	**4**	**1**	**–2**	**5**	**–2**
Y_q	**Y_1**	**Y_2**	**Y_3**	**Y_4**	**Y_5**
4	**5**	**2**	**1**	**–2**	**–3**

2. Решение

Обозначения переменных			
Имя переменной	Тип переменной		Пояснение
	Fortran	**C**	
N, i	**INTEGER**	**INT**	Количество точек и порядковый номер точки
XQ,YQ	**REAL**	**FLOAT**	Абсцисса и ордината точки $Q(X_q,Y_q)$
X_i, Y_i	**REAL**	**FLOAT**	Координаты точки с номером i
RQ	**REAL**	**FLOAT**	Расстояние между $M_i(X_i,Y_i)$ и $Q(X_q,Y_q)$
RMIN	**REAL**	**FLOAT**	Наименьшее расстояние
K	**INTEGER**	**INT**	Порядковый номер искомой точки
X_k,Y_k	**REAL**	**FLOAT**	Координаты искомой точки

3. Ручной расчёт результатов

Представим заданные точки на плоскости (X, Y).



Исходные данные заданы в соответствии с условием задачи. Это означает, что среди заданных три точки [$M_1(4; 5)$, $M_2(1; 2)$ и $M_5(-2; -3)$] расположены в первом и третьем квадранте. Из них искомой (ближайшей к точке Q) является точка $M_2(1; 2)$. Точка $Q(-2;4)$ выбрана во втором квадранте (может лежать в любом из них).

Расстояние между текущей точкой $M_i(X_i, Y_i)$ и $Q(X_q,Y_q)$ вычисляется только для точек первого и третьего квадранта при выполнении условия $X_i \cdot Y_i > 0$ по формуле

$$RQ = \sqrt{(X_i - X_q)^2 + (Y_i - Y_q)^2}.$$

Точки первого и третьего квадранта				
Номер точки i	Абсцисса	Ордината	Расстояние до точки Q (–2; 4)	
1	4	5	$\sqrt{37}$	6,08
2	1	2	$\sqrt{13}$	3,61
5	-2	-3	$\sqrt{49}$	7

Искомой является точка первого квадранта $M_2(1;2)$ с номером **K = 2**, расположенная от точки $Q(-2;4)$ на расстоянии **RMIN = 3,61.**

4. Алгоритм

При решении задачи используется алгоритм вычисления наименьшего значения. Параметр цикла – номер точки **i**.

Для контроля и наглядности результатов в п.7 выводятся точки первого и третьего квадрантов. Из них осуществляется выбор искомой точки, ближайшей к $Q(X_q,Y_q)$.

Алгоритм состоит из следующих действий:

1. Ввод данных – координат точек $M_i(X_i, Y_i)$ и $Q(X_q,Y_q)$.
2. Вывод координат точек $M_i(X_i, Y_i)$ и точки $Q(X_q,Y_q)$.
3. Задание начального значения переменной RMIN = 1.0E38.
4. *Начало цикла по i.*
5. Проверка условия $X_i \cdot Y_i > 0$. Если условие выполнено, перейти к п.6, иначе – к п.10.
6. Вычисление расстояния RQ.
7. Вывод значений переменных X_i, Y_i и RQ.
8. Проверка условия RQ < RMIN. Если условие выполнено, перейти к п.9, иначе – к п.10.
9. Переменным присваиваются значения: RMIN = RQ, K = i.
10. *Конец цикла по i.*
11. Вывод координат искомой точки $M_k(X_k,Y_k)$ и RMIN.
12. Конец.

Схема алгоритма представлена на рис. 2.18 (стр. 51).

5. Листинг Basic-программы

Программный код

```
' Вычисление координат искомой точки
CLS
DIM X(5), Y(5)
DATA 5,-2,4
READ N, XQ, YQ
PRINT "ТОЧКА Q; "; " XQ ="; XQ; " YQ = "; YQ
DATA 4,5,1,2,-2,1,5,-2,-2,-3
' Ввод-вывод координат точек
PRINT TAB(4); "КООРДИНАТЫ ТОЧЕК:"
FOR I = 1 TO N ' Начало цикла
  READ X(I), Y(I)
  PRINT "X("; I; ")="; X(I);
  PRINT "  Y("; I; ")="; Y(I)
NEXT I             ' Конец цикла
RMIN = 1E+38
PRINT TAB(6); "ТОЧКИ 1 И 3 КВАДРАНТА"
FOR I = 1 TO N     ' Начало цикла
  IF X(I) * Y(I) > 0 THEN
    RQ = SQR((X(I) - XQ) ^ 2 + (Y(I) - YQ) ^ 2)
```

```
' Вывод координат точек 1 и 3 квадранта
    PRINT "X("; I; ") ="; X(I);
    PRINT " Y("; I; ") ="; Y(I); " RQ ="; RQ
    IF RQ < RMIN THEN RMIN = RQ : K = I
  END IF
NEXT I                ' Конец цикла
' Вывод координат искомой точки
 PRINT TAB(14); "ИСКОМАЯ ТОЧКА:"
 PRINT "X("; K; ") = ";X(K); " Y("; K; ") = ";Y(K);
 PRINT USING " RMIN = #.## "; RMIN
END
```

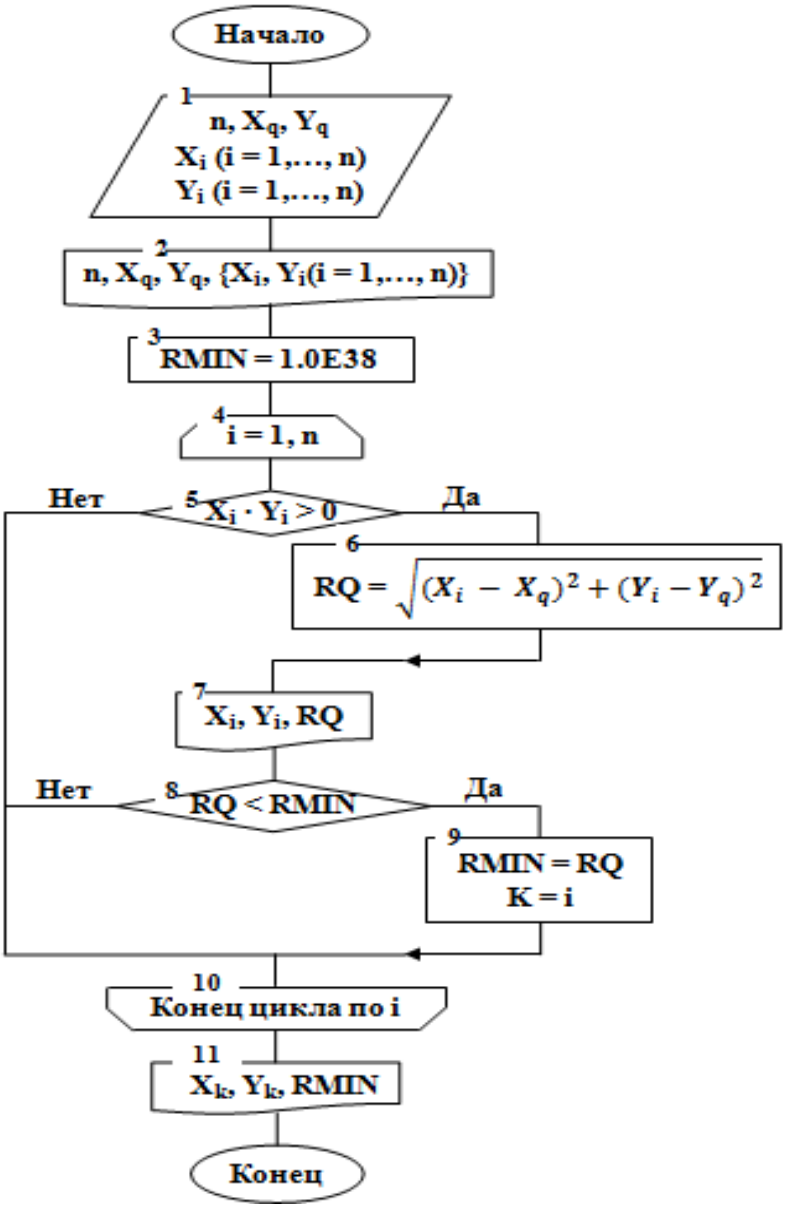


Рис. 2.18 Схема алгоритма к лабораторной работе 10 «Вычисление координат точки, ближайшей к заданной»

6. Макет печати результатов (по Basic-программе)

```
      ТОЧКА Q;  XQ = -2  YQ =  4
          КООРДИНАТЫ ТОЧЕК:
      X( 1 )= 4          Y( 1 )= 5
      X( 2 )= 1          Y( 2 )= 2
      X( 3 )=-2          Y( 3 )= 1
      X( 4 )= 5          Y( 4 )=-2
      X( 5 )=-2          Y( 5 )=-3
         ТОЧКИ 1 И 3 КВАДРАНТА
X( 1 ) = 4  Y( 1 ) = 5  RQ = 6.08
X( 2 ) = 1  Y( 2 ) = 2  RQ = 3.61
X( 5 ) =-2  Y( 5 ) =-3  RQ = 7
             ИСКОМАЯ ТОЧКА:
X( 2 ) = 1  Y( 2 ) = 2  RMIN = 3.61
```

Результаты ручного и машинного расчётов совпадают. В этом легко убедиться: искомая точка обозначена на графике (стр.49).

Лабораторная работа 11

Вычисление координат искомой точки

Тема. Одномерный массив: min, max

Индивидуальные варианты – табл. 3.11 (стр.114)

1. Задание

Составить программу решения следующей задачи.

На плоскости (x, y) заданы точки $\{M_1(X_1, Y_1), \ldots, M_n(X_n, Y_n)\}$, *n* – количество точек. Координаты точек задаются в виде массивов абсцисс $\mathbf{(X_1, \ldots, X_n)}$ и ординат $\mathbf{(Y_1, \ldots, Y_n)}$, *в соответствии с условием задачи* и так, чтобы *искомая точка была одна*.

Среди точек первого квадранта определить точку с наименьшей абсциссой, n = 5.

Для выполнения лабораторной работы необходимо:

- задать координаты точек $\{M_1(X_1, Y_1), \ldots, M_n(X_n, Y_n)\}$.
- представить заданные точки на плоскости (x,y).
- найти и обозначить на графике искомую точку.

Исходные данные

Абсциссы	X_1	X_2	X_3	X_4	X_5
	4	**1**	**–2**	**5**	**–2**
Ординаты	Y_1	Y_2	Y_3	Y_4	Y_5
	5	**2**	**1**	**–2**	**–3**

2. Решение

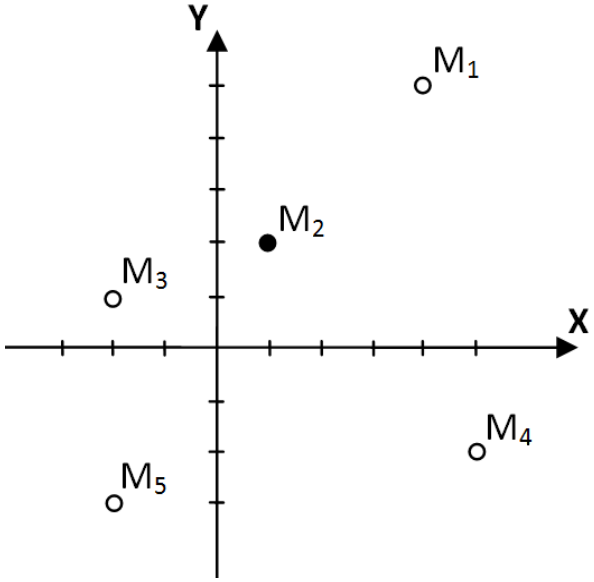
Обозначим: **K, XMIN** – номер и абсцисса искомой точки.

Обозначения переменных			
Имя переменной	Тип переменной		Пояснение
	Fortran	**C**	
N, i	**INTEGER**	**INT**	Количество точек и номер точки
X_i, Y_i	**REAL**	**FLOAT**	Координаты точки с номером i
XMIN	**REAL**	**FLOAT**	Абсцисса искомой точки
K	**INTEGER**	**INT**	Номер искомой точки
X_k,Y_k	**REAL**	**FLOAT**	Координаты искомой точки

3. Ручной расчёт результатов

Графическое представление точек:

Исходные данные заданы в соответствии с условием задачи. Это означает, что среди заданных две точки $M_1(4; 5)$ и $M_2(1; 2)$, расположены в первом квадранте. Из них искомой (с наименьшей абсциссой) является точка $M_2(1; 2)$. Принадлежность точки $M_i(X_i,Y_i)$ первому квадранту определяется условиями $X_i > 0$ и $Y_i > 0$.



При решении задачи используется алгоритм вычисления наименьшего значения. Параметр цикла – номер точки ***i*** .

Для контроля и наглядности результатов точки первого квадранта выводятся на печать. Из них осуществляется выбор искомой точки с наименьшей абсциссой XMIN.

Схема алгоритма представлена на рис. 2.19:

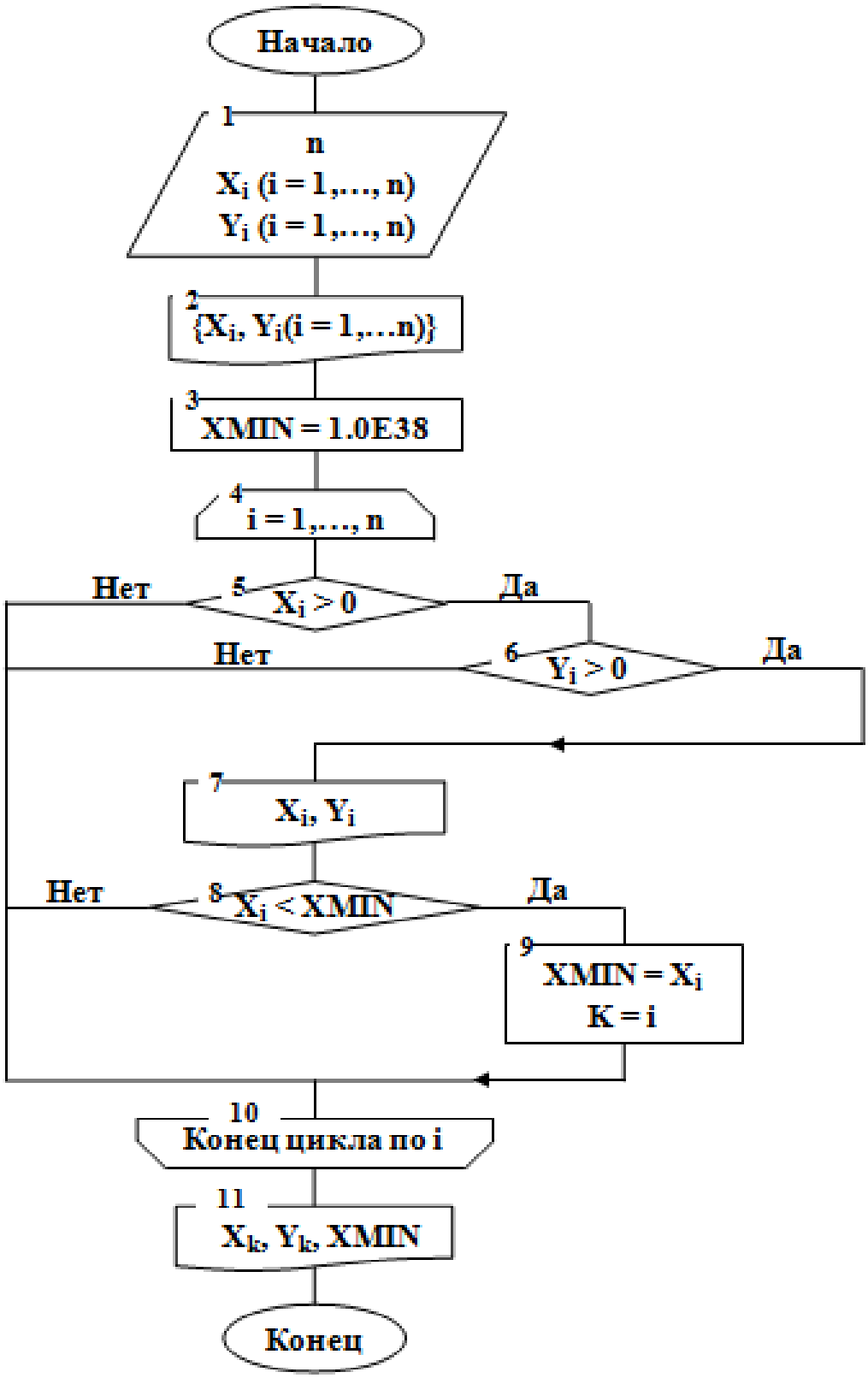


Рис. 2.19 Схема алгоритма к лабораторной работе 11 «Вычисление координат искомой точки»

4. Листинг Fortran-программы

*Условия **X(I) > 0** и **Y(I) > 0** объединены логической связкой* **.AND.** .

```
! Вычисление координат искомой точки
PROGRAM POINT
! Объявление типа и задание данных
INTEGER :: I,K,N = 5   ! I,K,N – целого типа
REAL :: XMIN = 1.0E38
REAL :: X(1 : 5) = (/4,1,-2,5,-2/)
REAL :: Y(1 : 5) = (/5,2,1,-2,-3/)
! Вывод координат точек
WRITE(*,1) '    КООРДИНАТЫ ТОЧЕК:'
WRITE(*,"(5F5.0/)")X,Y
WRITE(*,*)' ТОЧКИ, ЛЕЖАЩИЕ В 1 КВАДРАНТЕ: '
DO I = 1,N    ! Начало цикла
! Вычисление и вывод точек 1 квадранта
  IF(X(I) > 0 .AND. Y(I) > 0)THEN ! Если Xi>0 и Yi>0
    WRITE(*,2)I,X(I),I,Y(I)
    2 FORMAT(1X,'X(',I2,') = ',F3.0,    &
             1X,'Y(',I2,') = ',F3.0)
! Вычисление наименьшей абсциссы
    IF(X(I) < XMIN) THEN ! Если Xi < Xmin
      XMIN = X(I); K = I
    END IF
  END IF
END DO ! Конец цикла
! Вывод координат искомой точки Xk,Yk
WRITE(*,*)' ИСКОМАЯ ТОЧКА:'
WRITE(*,12)K,X(K),K,Y(K),XMIN
12 FORMAT(1X,'X(',I2,') = ',F3.0, 1X,    &
 'Y(',I2,') = ',F3.0,' XMIN = ',F3.0)
END
```

5. Макет печати результатов

```
      КООРДИНАТЫ ТОЧЕК:
  4.    1.   -2.    5.   -2.
  5.    2.    1.   -2.   -3.
 ТОЧКИ, ЛЕЖАЩИЕ В 1 КВАДРАНТЕ:
   X (1) =  4.  Y (1) =  5.
   X (2) =  1.  Y (2) =  2.
       ИСКОМАЯ ТОЧКА:
X (2) = 1. Y (2) = 2. XMIN = 1
```

Результаты ручного и машинного расчётов совпадают. На графике искомая точка обозначена (стр.53).

Лабораторная работа 12

Вычисление значения выражения

Тема. Одномерный массив: сумма, произведение, min, max

Индивидуальные задания – табл. 3.12 (стр.115)

Математические функции – табл. 5.1 (стр.136)

1. Задание

Дан массив $\mathbf{A_i}$, **(i = 1,2,…,n), n = 4**. Составить программу вычисления значения величины **Z**, заданной выражением:

$$Z = \sum_{i=2}^{n} i \bullet A_i + \max_i \left(A_i \bullet \prod_{i=1}^{n-1} i^2 \right)$$

При вычислении **max** значение ***i*** изменяется в интервале **[1, *n*]**.

Исходные данные

Количество элементов	**Элементы массива**			
n	$\mathbf{A_1}$	$\mathbf{A_2}$	$\mathbf{A_3}$	$\mathbf{A_4}$
4	**1,5**	**–2**	**3**	**1,2**

2. Решение

Введём дополнительные обозначения:

$$\mathbf{S} = \sum_{i=2}^{n} i \bullet A_i \ , \quad \mathbf{P} = \prod_{i=1}^{n-1} i^2 , \quad \mathbf{AMAX} = \max_i \left(A_i \bullet \prod_{i=1}^{n-1} i^2 \right), \quad \mathrm{R} = \mathbf{A_i} \cdot \mathbf{P}.$$

Тогда выражение для **Z** можно представить: **Z = S + AMAX**;

Обозначения переменных			
Имя переменной	Тип переменной		Пояснение
	Fortran	**C**	
i	**INTEGER**	**INT**	Порядковый номер элемента в массиве
$\mathbf{A}_i$	**REAL**	**FLOAT**	Элемент массива с номером i
N	**INTEGER**	**INT**	Количество элементов в массиве
S, P	**REAL**	**FLOAT**	Сумма и произведение
R	**REAL**	**FLOAT**	Рабочая переменная
AMAX	**REAL**	**FLOAT**	Наибольшее значение
Z	**REAL**	**FLOAT**	Искомая величина, результат

3. Ручной расчёт результатов

Вычислим значение Z в следующем порядке:

$$\mathbf{S} = \sum_{i=2}^{4} i \bullet A_i = 2\cdot A_2 + 3\cdot A_3 + 4\cdot A_4 = \mathbf{9,8} \qquad P = \prod_{i=1}^{3} i^2 = 1^2\cdot 2^2\cdot 3^2 = 36$$

$$\mathbf{AMAX} = \max_i \left(A_i \bullet P\right) = \max\left(A_1\cdot P, A_2\cdot P, A_3\cdot P, A_4\cdot P\right) = \mathbf{108}$$

$$\mathbf{Z} = S + AMAX = 9{,}8 + 108 = \mathbf{117,8}$$

Алгоритм содержит три цикла с параметром i. Схема представлена на рис. 2.20 (стр.58).

4. Листинг С-программы «Вычисление значения выражения»

Внимание. *При вводе программы следует иметь в виду, что С различает регистр букв, т.е., «**x**» и «**X**» – это две разные переменные. Нижняя граница индекса **i** (порядкового номера элемента в массиве) задаётся равной нулю.*

```
# include<stdio.h> // Заголовочные файлы
# include<math.h>
# include<conio.h>
  void main()
{
int i,n = 4; // Объявление типа и задание данных
 float s = 0, p = 1, amax = -1.0e38, r, z;
 float a[4] = {1.5,-2,3,1.2};
 printf("  \t   Array: \n");
 for(i = 0;i<n;i++) // Цикл по i
   printf(" %6.1f ",a[i]); // Вывод A(i)
 for(i = 1;i<n;i++) // Цикл вычисления S
   s+=(i+1)*a[i];
 for(i = 1;i<=n-1;i++) // Цикл вычисления P
   p = p*i*i;
 for(i = 0;i<n;i++) // Цикл вычисления AMAX
   {
    r = a[i] * p;
    if(r > amax) amax = r;
   }
 z = s + amax;
 printf(" \n \n \t  Result \n");
 printf("S = %.1f P = %.0f  AMAX = %.0f\n",s,p,amax);
 printf(" \n \t Z = %.1f ",z);
 getch();
 return;
 }
```

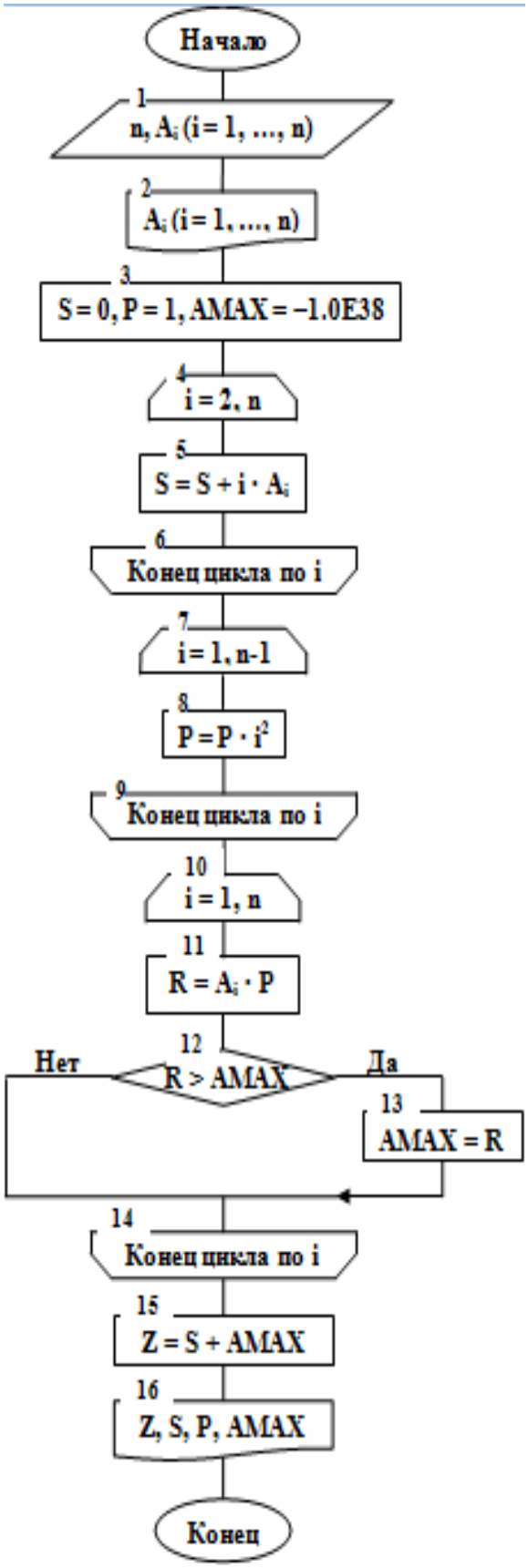
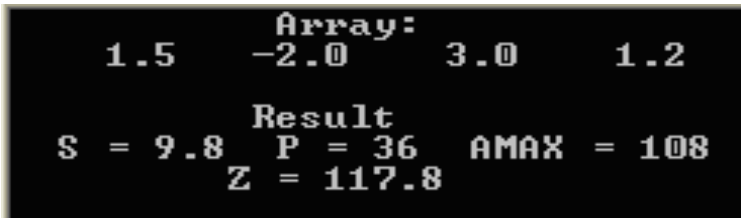


Рис. 2.20 Схема алгоритма к лабораторной работе 12 «Вычисление значения выражения»

5. Макет печати результатов

Как видно, результаты ручного и машинного расчётов совпадают.



Лабораторная работа 13

Анализ эффективности работы предприятия

Тема. Одномерный массив: сумма, количество, min, max

Индивидуальные варианты – табл. 3.13 (стр.117)

1. Задание

Составить программу решения следующей задачи:

Выполнение плана (в процентах) по объемам производства на судостроительном предприятии в течение года и названия месяцев представлены массивами $\mathbf{A_i}$ и $\mathbf{MC_i}$ **(i = 1,...,12)**. План – 100%, менее 100% – план не выполнен, более 100% – перевыполнен.

Определить среди месяцев с процентом выполнения плана выше среднего месяц с наименьшим процентом.

Исходные данные

1-е полугодие						
Месяц	1	2	3	4	5	6
	Январь	Февраль	Март	Апрель	Май	Июнь
%	$\mathbf{A_1}$	$\mathbf{A_2}$	$\mathbf{A_3}$	$\mathbf{A_4}$	$\mathbf{A_5}$	$\mathbf{A_6}$
	101	**98**	**100**	**108**	**97**	**105**
2-е полугодие						
Месяц	7	8	9	10	11	12
	Июль	Август	Сентябрь	Октябрь	Ноябрь	Декабрь
%	$\mathbf{A_7}$	$\mathbf{A_8}$	$\mathbf{A_9}$	$\mathbf{A_{10}}$	$\mathbf{A_{11}}$	$\mathbf{A_{12}}$
	99	**107**	**96**	**112**	**100**	**98**

2. Решение

Обозначим: **S**, **SR** – суммарный и средний процент выполнения плана. $\mathbf{S} = \sum_{i=1}^{12} A_i$, **SR = S / 12.**

K, AMIN – номер месяца с наименьшим процентом AMIN.

Обозначения переменных			
Имя переменной	Тип переменной		Пояснение
	Fortran	**C**	
I	**INTEGER**	**INT**	Номер элемента в массиве (месяца)
A_i	**REAL**	**FLOAT**	Элемент массива с номером I
MC_i	**CHARACTER**	**CHAR**	Элемент массива (месяц) с номером I
S, SR	**REAL**	**FLOAT**	Сумма и среднее арифметическое
AMIN	**REAL**	**FLOAT**	Наименьший процент выполнения плана
K	**INTEGER**	**INT**	Номер месяца с наименьшим процентом
MC_k	**CHARACTER**	**CHAR**	Название искомого месяца с номером K

3. Ручной расчёт результатов

Вычислим сумму **S** = 1221 и процент **SR** = S/12 = 101,75.

Месяцы с процентом выполнения плана выше среднегодового

Месяц	4	6	8	10
	Апрель	Июнь	Август	Октябрь
Процент выполнения плана	108	105	107	112

Очевидно, что наименьший процент **AMIN = 105** достигается в июне-месяце с порядковым номером **K = 6**.

4. Алгоритм

При решении задачи используется алгоритм вычисления суммы и наименьшего значения. Алгоритм содержит два цикла, параметр цикла – номер месяца **i**.

В первом цикле осуществляется вычисление суммы **S**, а во втором определяются и выводятся названия месяцев с процентом выше среднегодового в соответствии с условием $A_i > SR$, а также вычисляется номер месяца K с наименьшим процентом AMIN.

Алгоритм состоит из следующих действий:

1. Ввод исходных данных – значений A_i и MC_i (i = 1,…,12).
2. Вывод данных – процентов выполнения плана A_i.
3. Задание начального значения переменной S = 0,
   AMIN = 1.0E38 (заведомо большого числа).
4. *Начало цикла по i.*
5. Вычисление суммы $S = S + A_i$.

6. *Конец цикла.*
7. Вычисление среднегодового процента. SR = S / 12
8. *Начало цикла по i.*
9. Проверка условия A_i > SR.
   Если условие выполняется, перейти к п.10, иначе – к п.13.
10. Вывод i, MC_i, A_i – номера, названия месяца и процента.
11. Проверка условия A_i < AMIN.
    Если условие выполняется, перейти к п.12,иначе – к п.13.
12. Переменным присваиваются значения AMIN = A_i, K = i.
13. *Конец цикла.*
14. Вывод значений SR, K, MC_k, AMIN.
15. Конец.

Схема алгоритма представлена на рис. 2.21 (стр.63).

5. Листинг С-программы

Внимание. *При вводе программы следует иметь в виду, что С различает регистр букв, т.е., «**x**» и «**X**» – это две разные переменные. Нижняя граница индекса **i** (порядкового номера элемента в массиве) задаётся равной нулю.*

```
// Анализ итогов работы предпрятия
#include<stdio.h>  // Заголовочные файлы
#include<string.h>
#include<conio.h>
void main()
{
// Объявление типа и задание данных
 int i,k;
 float s = 0, amin = 1.0e38, sr;
float a[12] = {101,98,100,108,97,105,
                99,107,96,112,100,98};
 char*mc[12] = {"january","february","march","april",
            "may","june","july","august","september",
                 "october","november","december"};
 printf("  \t\t \t Percent \n");
 for(i = 0;i<12;i++) // Цикл по i
  {
   printf(" %.0f ",a[i]); // Вывод процентов Ai
   s += a[i]; // Вычисление суммы S
  }
```

```
 sr = s/12;
 printf(" \n sr =  %.2f \n ",sr); // Вывод SR
 printf("  N \t  Month  \t Percent \n ");
 for(i = 0;i<12;i++) // Цикл по i
    {
     if(a[i] > sr ) // Если A_i > SR
      {
 printf("i = %2d  %s  \t  %3.0f \n",i+1,mc[i],a[i]);
 if(a[i] < amin) // Если A_i < AMIN
        {
         amin = a[i]; k = i;
        }
      }
    }
printf(" \t Result \n"); // Вывод значений K,AMIN
 printf("k = %d  %s  amin = %.0f ",k+1,mc[k],a[k]);
 getch();
 return;
}
```

6. Макет печати результатов

```
                        Percent
 101  98  100  108  97  105  99  107  96  112  100  98
 sr =  101.75
  N       Month          Percent
i =  4     april           108
i =  6     june            105
i =  8     august          107
i = 10     october         112
         Result
k = 6    june   amin = 105
```

Как видно, результаты ручного и машинного расчётов совпадают.

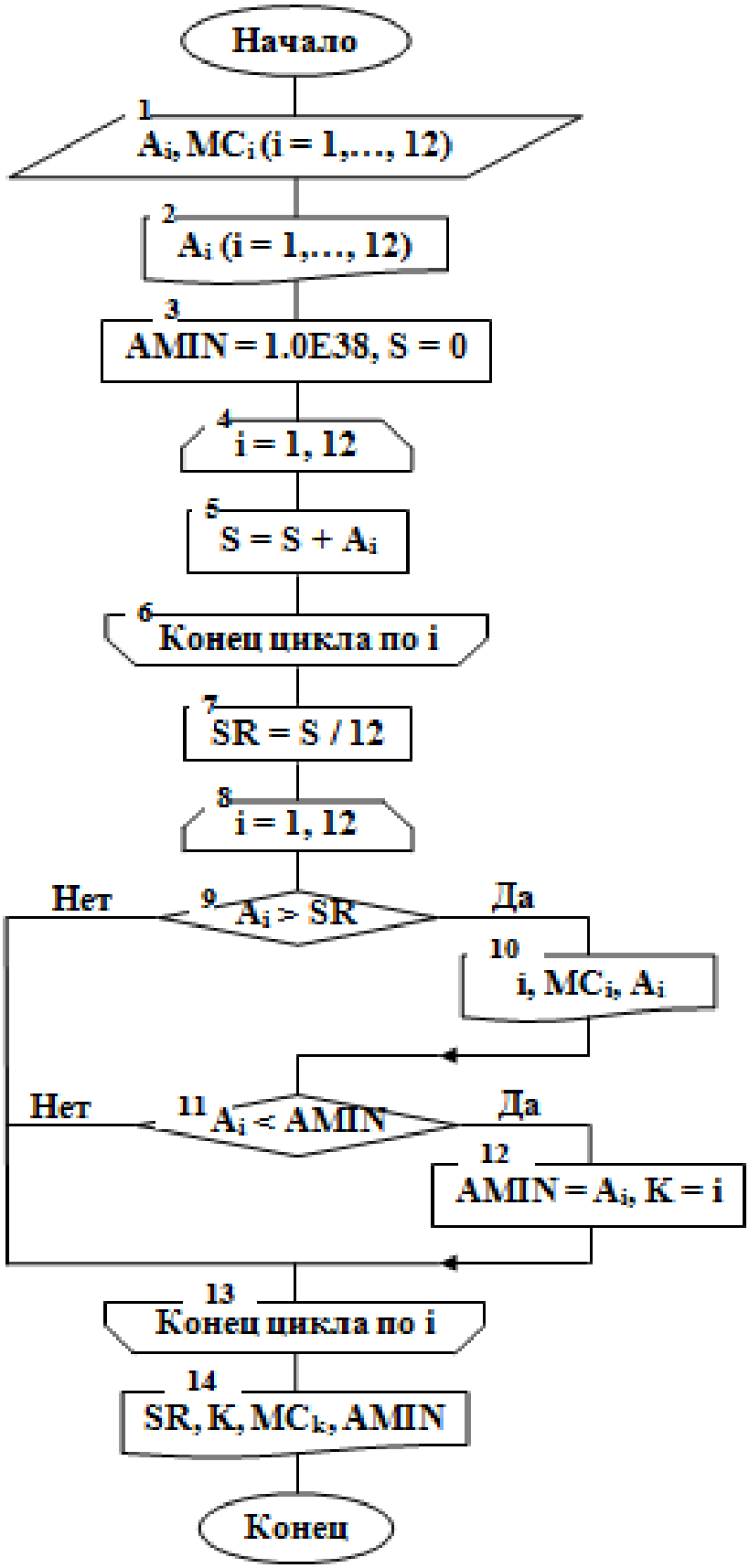


Рис. 2.21 Схема алгоритма к лабораторной работе 13 «Анализ эффективности работы предприятия»

Лабораторная работа 14

Анализ итогов экзамена в группе

Тема. Одномерный массив: сумма, количество

Индивидуальные задания – табл. 3.14 (стр.119)

1. Задание

Составить программу решения следующей задачи:

Результаты сдачи экзамена и фамилии студентов группы представлены массивами $\mathbf{A_i}$ и $\mathbf{F_i}$ **(i = 1,…,10)**.

Исходные данные

Студент	F_1	F_2	F_3	F_4	F_5
Фамилия	Юдин И.В.	Ашин С.В.	Итин К.Л.	Бут С.С.	Бех Л.Л.
Оценка	A_1	A_2	A_3	A_4	A_5
	5	4	4	3	2
Студент	F_6	F_7	F_8	F_9	F_{10}
Фамилия	Этин П.Н.	Юров Ю.Ю.	Апин Р.П.	Фет В.И.	Кац Д.Е.
Оценка	A_6	A_7	A_8	A_9	A_{10}
	5	4	2	4	3

Примечание. При вводе фамилий в программе инициалы не учитываются!

Определить количество студентов-хорошистов, а также количество и фамилии отличников.

2. Решение

Обозначения переменных			
Имя переменной	Тип переменной		Пояснение
	Fortran	**C**	
i	**INTEGER**	**INT**	Номер элемента (студента в группе)
$\mathbf{A_i}$	**REAL**	**FLOAT**	Элемент массива (оценка) с номером I
$\mathbf{F_i}$	**CHARACTER**	**CHAR**	Элемент (фамилия студента) с номером I
K4, K5	**INTEGER**	**INT**	Количество оценок «хорошо» и «отлично»

3. Ручной расчёт результатов

Подсчитаем количество оценок «хорошо» и «отлично». Очевидно, что K4 = 4, а K5 = 2.

4. Алгоритм

Вычисление количества хороших и отличных оценок в алгоритме осуществляется проверкой условий $A_i = 4$ и $A_i = 5$.

Алгоритм состоит из следующих действий:

1. Ввод данных – элементов массивов F_i, A_i (фамилий и оценок).
2. Вывод данных (оценок студентов) A_i.
3. Задание начального значения переменным K4 = 0 и K5 = 0.
4. *Начало цикла.*
5. Проверка условия $A_i = 5$.
   Если условие выполнено, перейти к п.6, иначе – к п.8.
6. Увеличить на 1 счётчик K5: K5 = K5 + 1.
7. Вывод фамилии студента-отличника F_i. Перейти к п.10.
8. Проверка условия $A_i = 4$.
   Если условие выполнено, перейти к п.9, иначе – к п.10
9. Увеличить на 1 значение счётчика K4: K4 = K4 + 1.
10. *Конец цикла.*
11. Вывод результатов – значений K5 и K4.
12. Конец.

Схема алгоритма представлена на рис. 2.22 (стр.66).

5. Листинг Fortran-программы «Анализ итогов экзамена»

```
PROGRAM MARKS
!  Объявление типа и задание данных
INTEGER :: I, K4 = 0, K5 = 0, N = 10
INTEGER :: A(1:10) = (/5,4,4,3,2,5,4,2,4,3/)
CHARACTER*8 :: F (1:10)
DATA F/' Юдин ',' Ашин ',' Итин ',' Бут ',' Бех ', &
       ' Этин ',' Юров ',' Апин ',' Фет ',' Кац '/
WRITE(*,*)' ОЦЕНКИ '
WRITE(*,' (2X,10I3) ') A ! Вывод оценок
WRITE(*,*)' ОТЛИЧНИКИ ' ! Вывод заголовка
DO I = 1 TO N ! Начало цикла
   IF(A(I) == 5)THEN ! Если Ai = 5
      K5 = K5 + 1
      WRITE(*,'(8X,A) ') F(I) ! Вывод отличников Fi
   ELSE ! Иначе, если Ai ≠ 5
      IF(A(I) == 4)K4 = K4 + 1 ! Если Ai = 4
   END IF
END DO ! Конец цикла
```

```
! Вывод значений K4, K5
WRITE (*, 2) K5,K5*100.0/N
2 FORMAT(2X,'K5 = ',I1,' % = ', F4.1)
WRITE (*, 3) K4,K4*100.0/N
3 FORMAT(2X,'K4 = ',I1,' % = ', F4.1)
END
```

Результаты работы программы представлены в п.7.

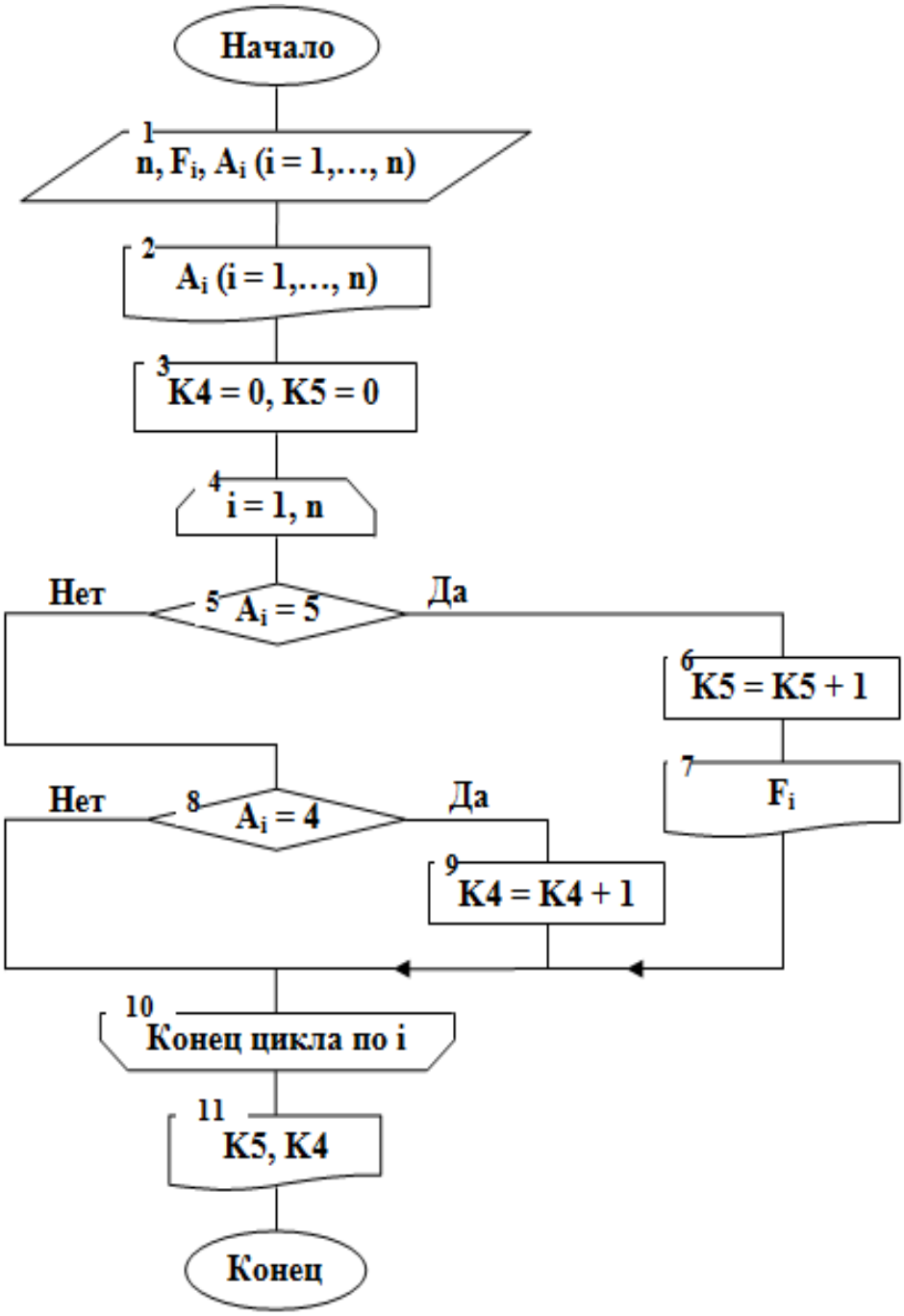


Рис. 2.22 Схема алгоритма к лабораторной работе 14 «Анализ итогов сдачи сессии»

6. Листинг Basic-программы

Примечание. При вводе фамилий в программе инициалы не учитываются!

```
' Анализ итогов сессии
CLS
DIM A(10), F$(10)
' Задание фамилий и оценок студентов
DATA "ЮДИН",5,"АШИН",4,"ИТИН",4,"БУТ",3,"БЕХ",2,
DATA "ЭТИН",5,"ЮРОВ",4,"АПИН",2,"ФЕТ",4,"КАЦ",3
PRINT TAB(7); " ОЦЕНКИ "
' Ввод фамилий и оценок
FOR I = 1 TO 10 ' Начало цикла
   READ F$(I), A(I)
   PRINT A(I); ' Вывод оценок Ai
NEXT I ' Конец цикла
PRINT  ' Пропуск строки
PRINT STRING$(28, "-") ' Вывод строки символов
 K5 = 0: K4 = 0: N = 10
 PRINT TAB(7); " ОТЛИЧНИКИ "
FOR I = 1 TO N ' Начало цикла
  IF A(I) = 5 THEN ' Если Ai = 5
    K5 = K5 + 1: PRINT TAB(6); F$(I)
  ELSE ' Иначе, если Ai ≠ 5
    IF A(I) = 4 THEN K4 = K4 + 1
  END IF
NEXT I ' Конец цикла
' Вывод результатов
PRINT USING " K5 = #  ## % "; K5, K5*100/N
PRINT USING " K4 = #  ## % "; K4, K4*100/N
END
```

Значения переменных K4 и K5 выводятся также и в процентах.

7. Макет печати результатов (по Basic-программе)

Как видно, результаты ручного и машинного расчётов совпадают.

```
                ОЦЕНКИ
5   4   4   3   2   5   4   2   4   3
---------------------------------------
              ОТЛИЧНИКИ
                ЮДИН
                ЭТИН
              K5 = 2 20%
              K4 = 4 40%
```

Контрольное задание для проверки знаний

Вычисление площади шпангоута

Тема. Одномерные массивы: сумма

Составить схему и программу решения для следующей задачи:

Для расчёта мореходных качеств судна необходимо знать элементы теоретического корпуса судна, имеющего сложную геометрическую форму. К таким элементам относят его объём, площади поперечных и продольных сечений, статические моменты объёма и площадей и др. Все выражения для вычисления элементов теоретического корпуса судна имеют интегральный вид. Например, площадь поперечного сечения судна, шпангоута (рис. 2.23), симметричного относительно ДП судна, определяется выражением:

$$\omega = 2\int_0^T y dz, \text{ где } y - \text{ординаты шпангоута.}$$

Контур половины шпангоута судна представлен на рис. 2.24.

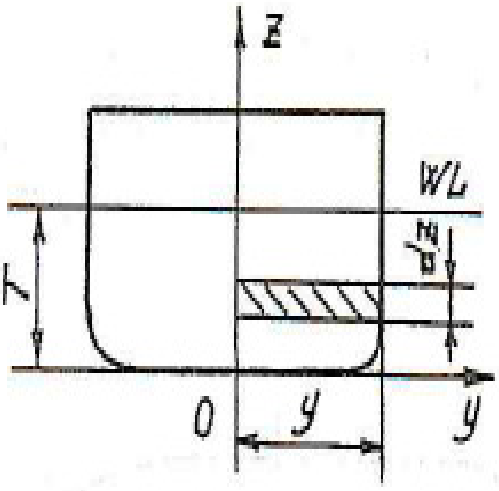


Рис. 2.23 «Контур шпангоута»

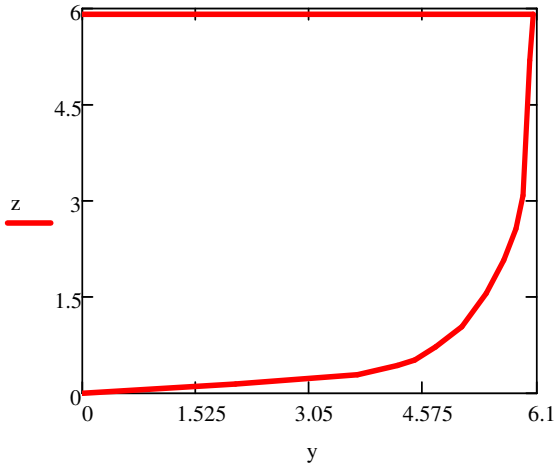


Рис. 2.24 «Контур половины шпангоута»

Однако все шпангоуты судна описываются кривыми y = f(z), для которых затруднительно подобрать аналитическое выражение. Поэтому расчёт площади ω проводят численным методом – по правилу трапеций с переменным шагом по формуле:

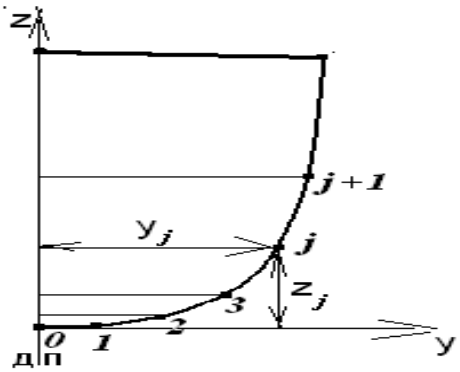
$$\omega = \sum_{j=1}^{m-1} f_j \ ,$$

где f_j – площадь j-той трапеции шпангоута.

$$f_j = \frac{1}{2}(z_{j+1} - z_j)(y_{j+1} + y_j)$$

где z_j, y_j – координаты j-той точки шпангоута, m – количество точек, j – номер точки. Они снимаются с теоретического чертежа судна в той последовательности, как показано на рис. 2.25.

Рис. 2.25 «Точки ветви шпангоута»



Задание

Вычислить площадь шпангоута методом трапеции по формуле:

$$\omega = \sum_{j=1}^{m-1} \frac{1}{2}(z_{j+1} - z_j)(y_{j+1} + y_j), \text{ где m = 13.}$$

Координаты ветви шпангоута представлены в таблице:

№	z	y
0	0	0
1	0,150	2,050
2	0,300	3,700
3	0,425	4,225
4	0,516	4,475
5	0,725	4,750
6	1,050	5,100
7	1,550	5,425
8	2,075	5,650
9	2,575	5,815
10	3,100	5,911
11	5,180	6,015
12	5,900	6,050
13	5,900	0

Лабораторная работа 15

Анализ эффективности работы танкерного флота

Тема. Двумерный массив (матрица)

Индивидуальные задания – табл. 3.15 (стр.120)

1. Задание

Составить схему и программу решения для следующей задачи.

Фактические данные по объёмам перевозок грузов пятью судами за четыре месяца навигации представлены в виде матрицы A(5x4), где строка – судно, столбец - месяц. Данные приведены в процентном отношении к нормативным объёмам перевозок (плану). За план принимается значение, равное 100%. Менее 100% – план не выполнен, более 100% – перевыполнен.

Матрица задаётся в соответствии с условием задачи. Элемент матрицы A_{ij} – это процент выполнения плана судном с номером i за месяц навигации с номером j.

Рассмотрим три варианта задачи:

Вариант 1.

Определить, каким судном и за какой месяц навигации достигнут наибольший процент выполнения плана.

Исходные данные

$$A = \begin{pmatrix} A_{11} & A_{12} & A_{13} & A_{14} \\ A_{21} & A_{22} & A_{23} & A_{24} \\ A_{31} & A_{32} & A_{33} & A_{34} \\ A_{41} & A_{42} & A_{43} & A_{44} \\ A_{51} & A_{52} & A_{53} & A_{54} \end{pmatrix} = \begin{pmatrix} 102 & 99 & 108 & 103 \\ 98 & 100 & 90 & 93 \\ 101 & 103 & 95 & 105 \\ 100 & 102 & 99 & 104 \\ 97 & 106 & 107 & 98 \end{pmatrix}$$

Названия судов и месяцев заданы массивами TN(5) и MC(4).

Суда					**Месяцы навигации**			
TN_1	TN_2	TN_3	TN_4	TN_5	MC_1	MC_2	MC_3	MC_4
Танкер–1	Танкер–2	Танкер–3	Танкер–4	Танкер–5	Май	Июнь	Июль	Август

Примечание. Возможен более простой и краткий вариант программы без использования названий судов и месяцев. Результаты представлены в макете печати (п.6, стр.74).

2. Решение

Математическая формулировка задачи:

Дана матрица A(5x4). Определить номер строки и столбца с наибольшим элементом матрицы.

Переменные, используемые в задаче, приведены в таблице:

Обозначения переменных			
Имя переменной	Тип переменной		Пояснение
	Fortran	**C**	
N, M	**INTEGER**	**INT**	Количество строк и столбцов матрицы
I, J	**INTEGER**	**INT**	Номер строки и столбца матрицы
$\mathbf{A_{ij}}$	**REAL**	**FLOAT**	Элемент матрицы
AMAX	**REAL**	**FLOAT**	Наибольший элемент матрицы
K, L	**INTEGER**	**INT**	Номер строки и столбца с наибольшим элементом
$\mathbf{TN_k}$, $\mathbf{MC_L}$	**CHARACTER**	**CHAR**	Название судна и месяца с наибольшим процентом выполнения плана

3. Ручной расчёт результатов

Процент **AMAX** = 108 достигнут судном «Танкер–1» с номером **K** = 1 за месяц навигации «июль» с номером **L** = 3.

4. Алгоритм

При решении задачи используется алгоритм вычисления наибольшего значения. Обход матрицы можно осуществлять и по строкам, и по столбцам. В алгоритме выбран вариант «по строкам». Алгоритм – двойной цикл вложенной структуры. Параметр внешнего цикла – номер строки **i** (судна), а внутреннего – номер столбца **j** (месяца).

Схема алгоритма представлена на рис. 2.26:

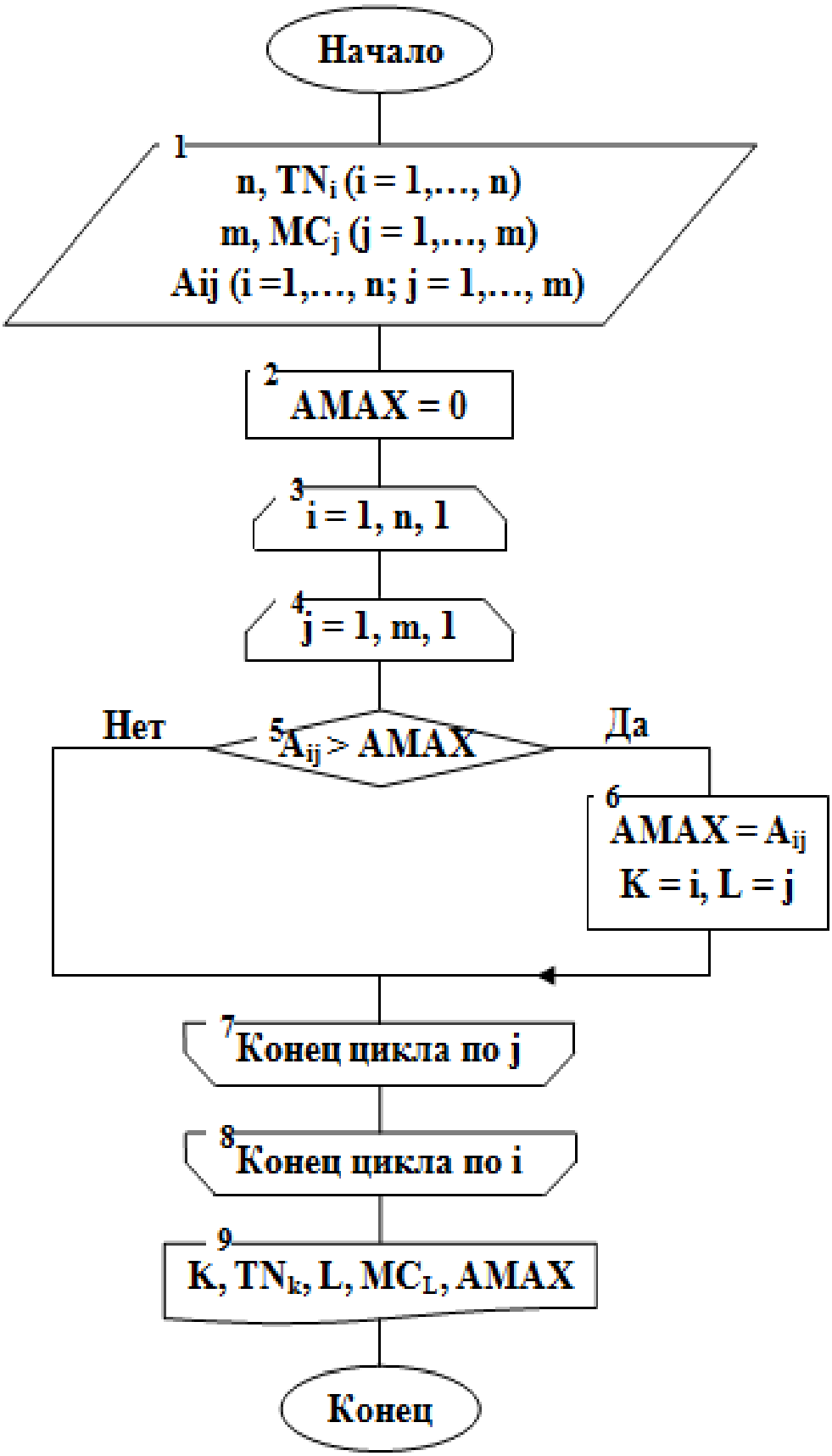


Рис. 2.26 Схема алгоритма к лабораторной работе 15 (вариант 1) «Анализ эффективности работы флота»

5. Листинг С-программы «Анализ работы флота»

Внимание. *При вводе программы следует иметь в виду, что С различает регистр букв, т.е., «**x**» и «**X**» – это две разные переменные. Нижние границы индексов **i** и **j** (номера строки и столбца) задаются равными нулю.*

```
#include<stdio.h>  // Заголовочные файлы
#include<conio.h>
#include<string.h>
 void main()
{
// Объявление типа и задание данных
 int i,j,k,l,n = 5,m = 4;
 char *mc[4] = {"May","June","July","August"};
 char *tn[5] = {"Tankep-1","Tankep-2",
          "Tankep-3","Tankep-4","Tankep-5"};
 float amax,a[5][4] = {102, 99,108,103,
                        98,100, 90, 93,
                       101,103, 95,105,
                       100,102, 99,104,
                        97,106,107, 98};
// Вывод названий месяцев – цикл по i
 printf("  Tankep");
 for(i = 0;i<m;i++)
   printf("  %s ",mc[i]);

 printf(" \n "); // Пропуск строки
   amax = 0;
 for(i = 0;i<n;i++) // Внешний цикл по i
  {
   printf("%s",tn[i]); // Вывод названия танкера TNi
   for(j = 0;j<m;j++) // Внутренний цикл по j
   {
    printf("%7.0f",a[i][j]); // Вывод процентов Aij
    if(a[i][j]>amax) // Если Aij > AMAX
      {
       amax = a[i][j];
       k = i; l = j;
      }
   }
 printf(" \n "); // Пропуск строки
  }
```

```
// Вывод результатов
 printf(" \n \t R e s u l t  \n ");
 printf(" Tankep  Month Percent \n");
 printf(" %s  %s %5.0f \n ", tn[k],mc[l],amax);
 printf(" K = %d L= %d AMAX =%.0f ",k+1,l+1,amax);
 getch();
 return;
}
```

6. Макет печати (для варианта 1)

Для сравнения приведён макет печати для варианта без названий судов и месяцев. Программу составить самостоятельно.

По С-программе

```
 Tankep     May   June   July   August
Tankep-1    102     99    108     103
Tankep-2     98    100     90      93
Tankep-3    101    103     95     105
Tankep-4    100    102     99     104
Tankep-5     97    106    107      98

        R e s u l t
 Tankep  Month Percent
Tankep-1  July   108
 K = 1   L = 3 AMAX=108
```

```
        МАТРИЦА
102     99    108    103
 98    100     90     93
101    103     95    105
100    102     99    104
 97    106    107     98
      Результат
     K = 1  L = 3
     AMAX = 108
```

Результаты ручного и машинного расчётов совпадают.

7. Листинг Basic-программы

Программный код	Пояснения

`' Анализ эффективности работы флота`
`CLS`
`DIM A(5, 4), MC$(4), TN$(5)`
`DATA "МАЙ  ", " ИЮНЬ  ", "ИЮЛЬ", " АВГУСТ "`
`DATA "Tankep-1 ", "Tankep-2","Tankep-3 "`
`DATA "Tankep-4 ","Tankep-5 "`
`N = 5 : M = 4`
	Задание месяцев и судов

`DATA 102,  99, 108, 103`
`DATA  98, 100,  90,  93`
`DATA 101, 103,  95, 105`
`DATA 100, 102,  99, 104`
`DATA  97, 106, 107,  98`
`PRINT "Tankep"; SPC(7);`
	Задание элементов матрицы

```
FOR I = 1 TO M    ' Начало цикла
 READ MC$(I)
 PRINT MC$(I);
NEXT I            ' Конец цикла
AMAX = 0
FOR I = 1 TO N ' Внешний цикл по I
  READ TN$(I)
  PRINT TN$(I);
  FOR J = 1 TO M ' Внутренний цикл по J
    READ A(I, J)
    PRINT USING "######"; A(I, J);
    IF A(I, J) > AMAX THEN
      AMAX = A(I, J)
      K = I: L = J
    END IF
  NEXT J  ' Конец внутреннего цикла по J
PRINT ' Пропуск строки
NEXT I        ' Конец внешнего цикла по I
PRINT TAB(10); "Результат"
PRINT "Танкер Месяц Процент"
PRINT TN$(K), MC$(L), AMAX
PRINT " K = ";K;" L = ";L;" AMAX = ";AMAX
END
```

Ввод-вывод месяцев

Ввод-вывод танкеров

Ввод-вывод строки

Вычисление AMAX, K, L

Вывод результатов

Результаты представлены в макете печати (п.6, стр.74).

Вариант 2.

1. Задание

Определить для каждого судна месяц навигации с наибольшим процентом выполнения плана.

Исходные данные

$$A = \begin{pmatrix} 102 & 99 & \mathbf{108} & 103 \\ 98 & \mathbf{100} & 90 & 93 \\ 101 & 103 & 95 & \mathbf{105} \\ 100 & 102 & 99 & \mathbf{104} \\ 97 & 106 & \mathbf{107} & 98 \end{pmatrix}$$

2. Решение

Математическая формулировка задачи:

Дана матрица A(5x4). Определить для каждой строки номер столбца с наибольшим элементом.

Таблица имён переменных приведена в варианте 1 (стр.71).

Здесь: **AMAX** – наибольший элемент строки матрицы (процент); **L, MC_L** – номер и название месяца с процентом AMAX.

3. Ручной расчёт результатов

Результаты расчёта приведены в следующей таблице:

Судно (строка)		**Месяц (столбец)**		**Наибольший элемент строки (процент AMAX)**
Номер i	**Название TN_i**	**Номер L**	**Название MC_L**	
1	Танкер–1	3	Июль	108
2	Танкер–2	2	Июнь	100
3	Танкер–3	4	Август	105
4	Танкер–4	4	Август	104
5	Танкер–5	3	Июль	107

4. Алгоритм

Алгоритм решения задачи – двойной цикл вложенной структуры. Обход матрицы для данного варианта осуществляется *только по строкам* (судам). Параметр внешнего цикла – номер строки **i** (судна), а внутреннего – номер столбца **j** (месяца).

Схема алгоритма представлена на рис. 2.27 (стр.80).

5. Макет печати (для варианта 2)

```
 Tankep    May   June   July  August  I Percent  Month
Tankep-1   102     99    108    103   I   108    July
Tankep-2    98    100     90     93   I   100    June
Tankep-3   101    103     95    105   I   105    August
Tankep-4   100    102     99    104   I   104    August
Tankep-5    97    106    107     98   I   107    July
```

Приведём продолжение С-программы варианта 2. Начало программы: файлы заголовков, объявление типа данных, названия судов, месяцев, матрицы – см. вариант 1 (п.5, стр.73).

6. Фрагмент листинга С-программы

Внимание. *При вводе программы следует иметь в виду, что С различает регистр букв, т.е., «**x**» и «**X**» – это две разные переменные. Нижние границы индексов **i** и **j** (номера строки и столбца) задаются равными нулю.*

```
 printf(" I Percent  Month \n ");
// Внешний цикл по i (по строкам)
 for(i = 0;i<n;i++)
   {
    amax = 0;
    printf("%s",tn[i]); // Вывод названия танкера
// Внутренний цикл по j (по столбцам)
    for(j = 0;j<m;j++)
     {
      printf("%6.0f",a[i][j]); // Вывод процентов Aij
// Вычисление мах элемента в строке AMAX и L
      if(a[i][j]>amax) // Если Aij > AMAX
       {
        amax = a[i][j]; l = j;
       }
     }
// Вывод для каждого танкера мах процента и месяца
 printf("    I %4.0f     %s \n ",amax,mc[l]);
   }
 getch();
 return;
 }
```

Результаты представлены в макете печати (п.5, стр.76).

Вариант 3.

1. Задание

Определить для каждого месяца навигации судно с наибольшим процентом выполнения плана.

Исходные данные

$$A = \begin{pmatrix} \mathbf{102} & 99 & \mathbf{108} & 103 \\ 98 & 100 & 90 & 93 \\ 101 & 103 & 95 & \mathbf{105} \\ 100 & 102 & 99 & 104 \\ 97 & \mathbf{106} & 107 & 98 \end{pmatrix}$$

2. Решение

Математическая формулировка задачи:

Дана матрица A(5x4). Определить для каждого столбца матрицы номер строки с наибольшим элементом.

Таблица имён переменных приведена в варианте 1 (стр.71).

Здесь: **AMAX** – наибольший элемент столбца матрицы (процент); **K, TN_k** – номер и название судна с процентом AMAX.

3. Ручной расчёт результатов

Результаты расчёта приведены в следующей таблице:

Месяц (столбец)		Судно (строка)		Наибольший элемент столбца матрицы (процент AMAX)
Номер j	Название MC(j)	Номер K	Название TN(k)	
1	Май	1	Танкер – 1	102
2	Июнь	5	Танкер – 5	106
3	Июль	1	Танкер – 1	108
4	Август	3	Танкер – 3	105

4. Алгоритм

Алгоритм решения задачи – двойной цикл вложенной структуры. Обход матрицы для данного варианта осуществляется *только по столбцам*. Параметр внешнего цикла – номер столбца j (месяца), а внутреннего – номер строки i (судна).

Схема алгоритма представлена на рис. 2.28 (стр.80).

Приведём продолжение С-программы варианта 3. Начало программы: файлы заголовков, объявление типа данных, названия судов, месяцев, задание матрицы – см. вариант 1 (п.5, стр.73).

5. Фрагмент листинга С-программы

Внимание. *При вводе программы следует иметь в виду, что С различает регистр букв, т.е., «**x**» и «**X**» – это две разные переменные. Нижние границы индексов **i** и **j** (номера строки и столбца) задаются равными нулю.*

```
 printf(" \n "); // Пропуск строки
// Внешний цикл по i (по строкам)
 for(i = 0;i<n;i++)
   {
    printf("%s",tn[i]); // Вывод названия танкера
```

```
// Внутренний цикл по j (по столбцам)
    for(j = 0;j<m;j++)
      printf("%7.0f",a[i][j]); // Вывод процентов
 printf(" \n "); // Пропуск строки
   }
// Вычисление для месяцев танкера с мах процентом
  printf(" \t Result \n Percent   Tankep  Month \n");
// Внешний цикл по j (столбцам)
  for(j = 0;j<m;j++)
    {
     amax = 0;
// Внутренний цикл  по i (строкам)
     for(i = 0;i<n;i++)
       {
        if(a[i][j]>amax) // Если Aij > AMAX
          {
           amax = a[i][j]; k = i;
          }
       }
// Вывод результатов для каждого месяца
    printf(" %5.0f   %s   %s  \n ",amax,tn[k],mc[j]);
    }
 getch();
 return;
}
```

6. Макет печати (для варианта 3)

```
  Tankep      May    June    July    August
Tankep-1      102      99     108      103
Tankep-2       98     100      90       93
Tankep-3      101     103      95      105
Tankep-4      100     102      99      104
Tankep-5       97     106     107       98
          Result
Percent    Tankep       Month
  102     Tankep-1      May
  106     Tankep-5      June
  108     Tankep-1      July
  105     Tankep-3      August
```

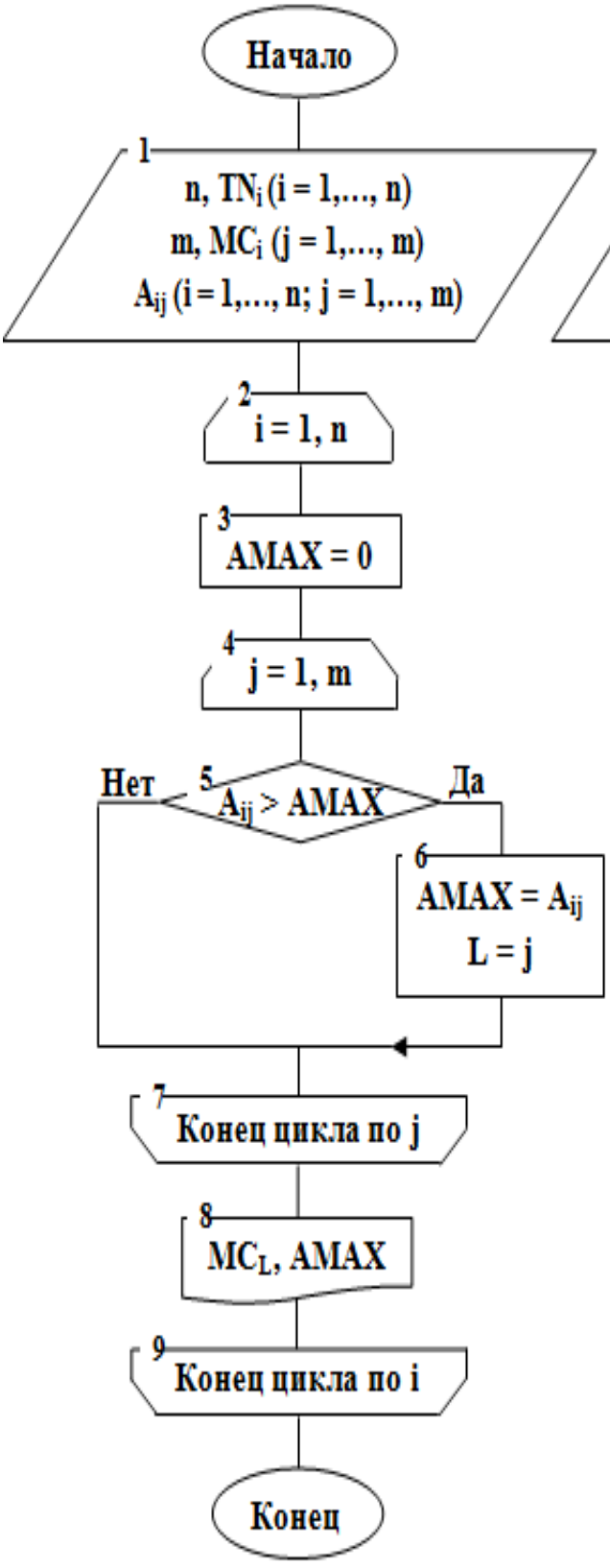


Рис. 2.27 Схема алгоритма к лабораторной работе 15 (вариант 2) «Анализ эффективности работы флота»

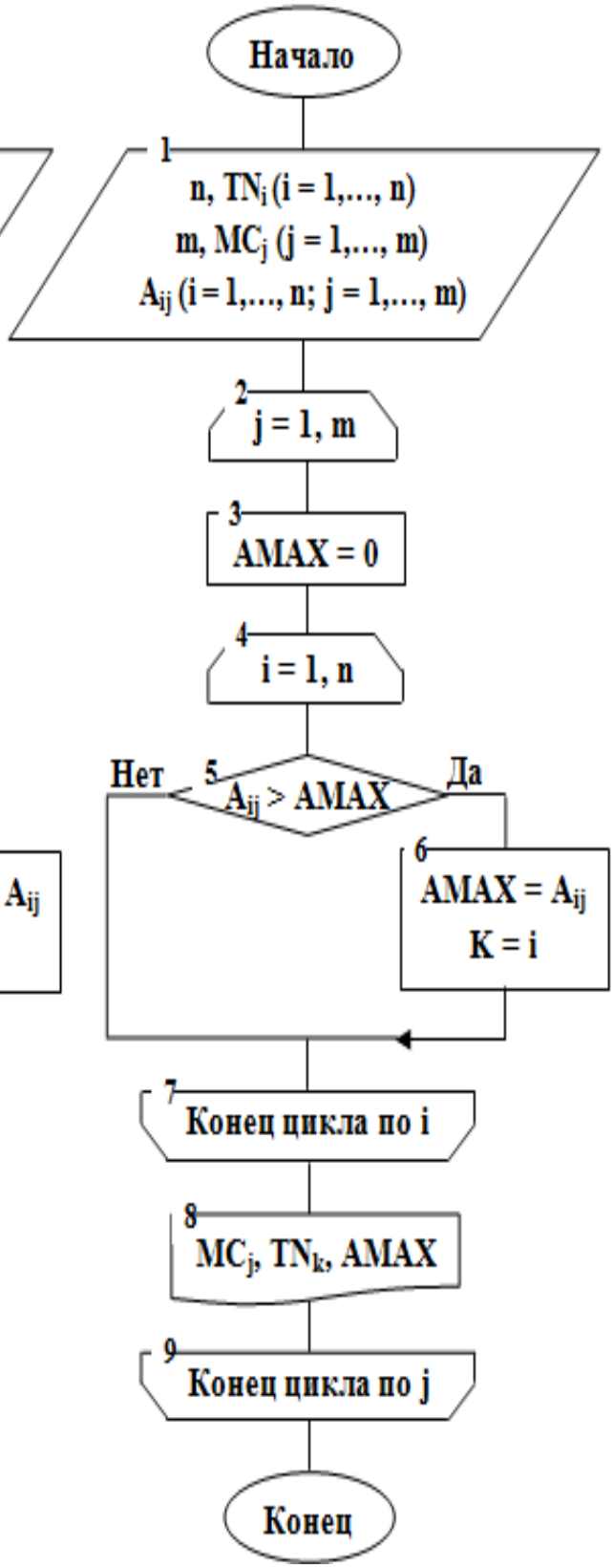


Рис. 2.28 Схема алгоритма к лабораторной работе 15 (вариант 3) «Анализ эффективности работы флота»

Лабораторная работа 16

Анализ итогов сессии в группе

Тема. Двумерный массив (матрица)

Индивидуальные задания – табл.3.16 (стр.122)

1. Задание

Составить схему и программу решения следующей задачи.

Результаты сессии из четырёх предметов в группе из пяти студентов представлены в виде матрицы A(5x4), где строка – студент, столбец – предмет. Элемент матрицы A_{ij} – оценка, полученная студентом с номером i по предмету с номером j. Матрица задаётся в соответствии с условием задачи.

Определить средний балл каждого студента и студента с наименьшим средним баллом.

Исходные данные

$$A = \begin{pmatrix} A_{11} & A_{12} & A_{13} & A_{14} \\ A_{21} & A_{22} & A_{23} & A_{24} \\ A_{31} & A_{32} & A_{33} & A_{34} \\ A_{41} & A_{42} & A_{43} & A_{44} \\ A_{51} & A_{52} & A_{53} & A_{54} \end{pmatrix} = \begin{pmatrix} 3 & 4 & 4 & 3 \\ 4 & 5 & 4 & 5 \\ 3 & 3 & 3 & 4 \\ 5 & 5 & 4 & 5 \\ 4 & 4 & 3 & 4 \end{pmatrix}$$

2. Решение

Математическая формулировка задачи:

Дана матрица A (5x4). Найти среднее арифметическое элементов каждой строки и номер строки с наименьшим средним арифметическим.

Переменные, используемые в задаче, приведены в таблице:

Обозначения переменных			
Имя переменной	Тип переменной		Пояснение
	Fortran	**C**	
I, J	**INTEGER**	**INT**	Номер текущей строки и столбца
A_{ij}	**INTEGER**	**INT**	Элемент матрицы (оценка студента)
S, SR	**REAL**	**FLOAT**	Суммарный и средний балл студента
SMIN	**REAL**	**FLOAT**	Наименьший средний балл
K	**INTEGER**	**INT**	Номер студента с баллом SMIN

3. Ручной расчёт результатов

Представим результаты расчёта в следующей таблице:

Сведения о студенте						
Номер строки i	**Оценки**				**Суммарный балл S**	**Средний балл SR**
1	**3**	**4**	**4**	**3**	14	3,5
2	**4**	**5**	**4**	**5**	18	4,5
3	**3**	**3**	**3**	**4**	13	3,25
4	**5**	**5**	**4**	**5**	19	4,75
5	**4**	**4**	**3**	**4**	15	3,75

Балл **SMIN** = 3,25 получил студент с номером **K** = 3.

4. Алгоритм

При решении задачи используется алгоритм вычисления суммы и наименьшего значения. Алгоритм содержит два цикла вложенной структуры. Обход матрицы осуществляется только по строкам. Параметр внешнего цикла – номер строки **i** (студента), а внутреннего – номер столбца **j** (предмета).

Схема алгоритма приведена на рис. 2.29. Результаты представлены в макете печати (п.5).

5. Макет печати результатов

По C-программе

```
 Session results    SR

  3   4   4   3  I 3.50

  4   5   4   5  I 4.50

  3   3   3   4  I 3.25

  5   5   4   5  I 4.75

  4   4   3   4  I 3.75

Minimum: SMIN =  3.25  K =  3
```

По Fortran-программе

```
      ОЦЕНКИ              SR

  3    4    4    3      3.50
  4    5    4    5      4.50
  3    3    3    4      3.25
  5    5    4    5      4.75
  4    4    3    4      3.75

    SMIN = 3.25 K = 3
```

Результаты ручного и машинного расчётов совпадают.

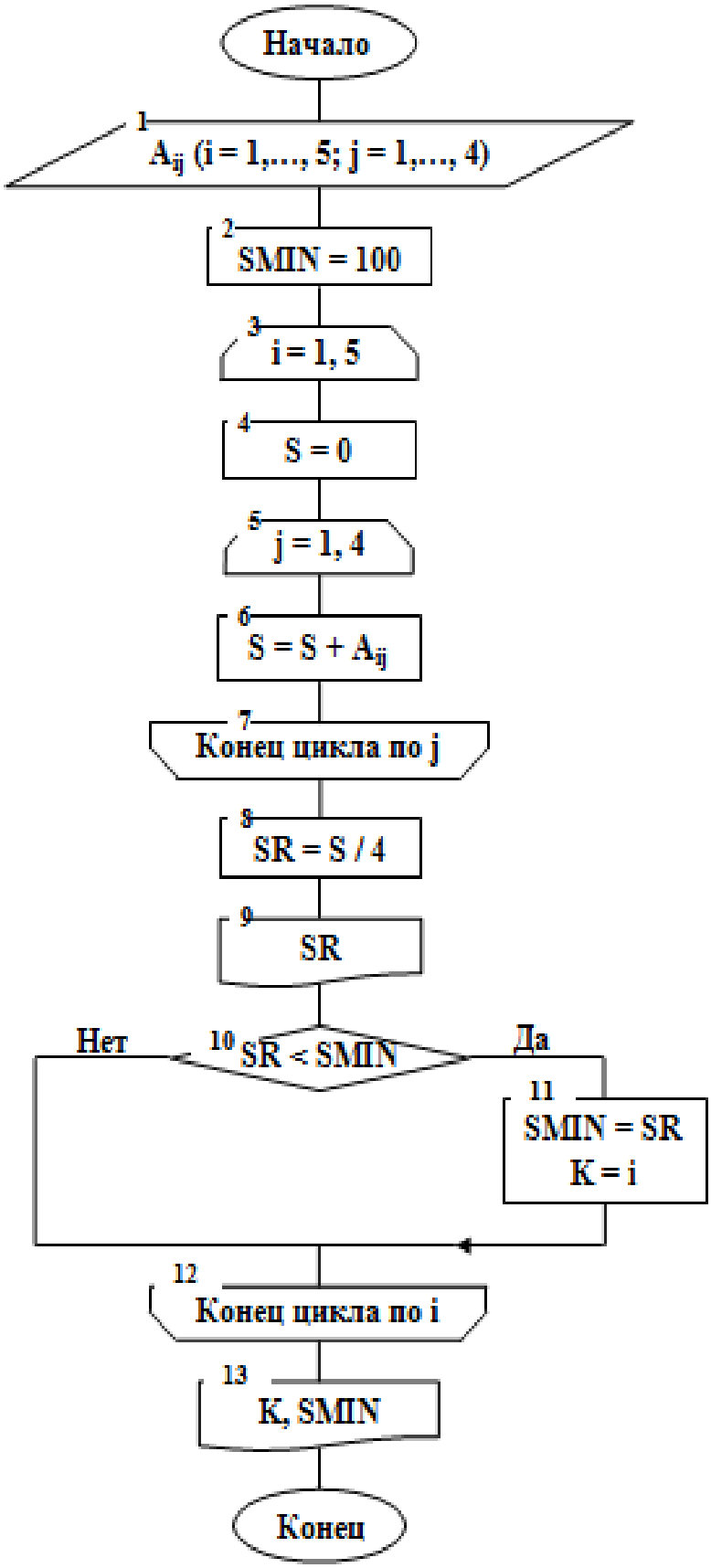


Рис. 2.29 Схема алгоритма к лабораторной работе 16 «Анализ итогов сессии в группе»

6. Листинг С-программы «Анализ итогов сессии в группе»

Внимание. *При вводе программы следует иметь в виду, что С различает регистр букв, т.е., «**x**» и «**X**» – это две разные переменные. Нижние границы индексов **i** и **j** (номера строки и столбца) задаются равными нулю.*

```
#include<stdio.h> // Заголовочные файлы
#include<conio.h>
 void main()
{
 int i,j,k; // Объявление типа данных
 float s, sr, smin = 100;
// Задание матрицы A(5,4) – оценок (по строкам)
 float a[5][4] = {3,4,4,3,4,5,4,5,3,3,
                  3,4,5,5,4,5,4,4,3,4};
 printf("  Session results    SR \n  ");
// Внешний цикл по строкам (студентам)
 for(i = 0;i<5;i++)
   {
    printf(" \n ");  // Пропуск строки
    s = 0;
// Внутренний цикл по столбцам (предметам)
      for(j = 0;j<4;j++)
        {
         s = s + a[i][j];
         printf(" %2.0f ",a[i][j]); // Вывод Aij
        }
    sr = s/4;
    printf(" I %.2f \n",sr);  // Вывод балла SR
// Вычисление минимального балла SMIN и K
    if(sr<smin) // Если SR < SMIN
      {
       smin = sr;
       k = i;
      }
   }
// Вывод SMIN и K
 printf(" Minimum: SMIN = %.2f K = %d ",smin,k+1);
 getch();
 return;
 }
```

7. Листинг Fortran-программы «Анализ итогов сессии»

Примечание. Матрица задаётся в файле с именем **FORT.1**!

Программный код

```
PROGRAM SESSION
! Объявление типа данных
INTEGER :: I,J,K,A(1:5,1:4)
REAL :: S,SR,SMIN = 100
! Ввод матрицы по строкам из файла < FORT.1 >
READ(1,*)((A(I,J),J = 1,4),I = 1,5)
WRITE(*,*)'  ОЦЕНКИ    SR '
DO I = 1,5 ! Внешний цикл по I (по строкам)
   S = 0.
   DO J = 1,4 ! Внутренний цикл по J (по столбцам)
     S = S + A(I,J)
   END DO ! Конец цикла по J
   SR = S/4
   ! Вывод строки матрицы и балла SR
   WRITE(*,3)(A(I,J),J = 1,4),SR
   3 FORMAT(2X,4I4,F4.2)
   ! Вычисление балла SMIN и номера K
   IF(SR < SMIN)THEN ! Если SR < SMIN
     SMIN = SR; K = I
   END IF
END DO ! Конец цикла по I
! Вывод значений SMIN и K
WRITE(*,"(A,F4.2,A,I1)")' SMIN = ',SMIN,' K = ',K
END
```

Результаты работы представлены в макете печати (п.5, стр.82).

Лабораторная работа 17

Формирование одномерного массива

Тема. Двумерный и одномерный массивы

Индивидуальные задания – табл.3.17 (стр.123)

1. Задание

Составить программу решения следующей задачи.

Дана матрица A(3x3). Сформировать одномерный массив (B_1,..., B_3) в соответствии с условием: **каждый элемент массива определяется разностью между элементом главной диагонали и количеством отрицательных элементов строки. Найти произведение положительных элементов массива B.**

Исходные данные

$$A = \begin{pmatrix} A_{11} & A_{12} & A_{13} \\ A_{21} & A_{22} & A_{23} \\ A_{31} & A_{32} & A_{33} \end{pmatrix} = \begin{pmatrix} 1 & -2 & -3 \\ 2 & 6 & -4 \\ -1 & 3 & 2 \end{pmatrix}$$

2. Решение

Переменные, используемые в задаче, приведены в таблице:

Обозначения переменных			
Имя переменной	Тип переменной		Пояснение
	Fortran	**C**	
I, J	**INTEGER**	**INT**	Номер строки и столбца матрицы
A_{ij}	**INTEGER**	**INT**	Элемент матрицы
K	**INTEGER**	**INT**	Количество отрицательных элементов строки
B_i	**INTEGER**	**INT**	Элемент одномерного массива B с номером i
P	**INTEGER**	**INT**	Произведение положительных элементов массива B

3. Ручной расчёт результатов

С учётом обозначений элементы массива вычисляются по формуле: **$B_i = A_{ii} – K$.**

Представим результаты расчёта в таблице:

Вычисление элементов массива B_i			
Номер строки i	Элемент главной диагонали A_{ii}	Количество K	Значение B_i
1	**1**	**2**	**–1**
2	**6**	**1**	**5**
3	**2**	**1**	**1**

Произведение положительных элементов массива **P = 5.**

4. Алгоритм

При решении используется алгоритм вычисления количества и произведения. Алгоритм – двойной цикл вложенной структуры. Обход матрицы осуществляется *только по строкам.* Параметр внешнего цикла – номер строки **i**, а внутреннего – номер столбца **j**.

Схема алгоритма представлена на рис. 2.30 (стр.89).

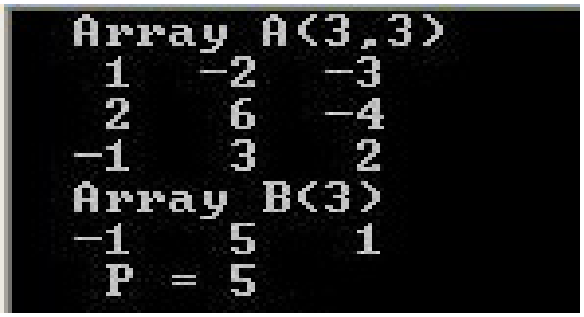
5. Листинг С-программы «Формирование массива»

Внимание. *При вводе программы следует иметь в виду, что С различает регистр букв, т.е., «**x**» и «**X**» – это две разные переменные. Нижние границы индексов **i** и **j** (номера строки и столбца) задаются равными нулю.*

```
#include<stdio.h> // Заголовочные файлы
#include<conio.h>
    void main()
{
 int i,j,k,n = 3; // Объявление типа данных
 float b[3],p = 1;
// Задание матрицы A по строкам
 float a[3][3] = {1,-2,-3,
                  2, 6,-4,
                 -1, 3, 2};
 printf(" \t ARRAY A(3,3):");
// Внешний цикл по строкам I
 for(i = 0;i < n;i++)
   {
    printf(" \n ");  // Пропуск строки
    k = 0;
// Внутренний цикл по столбцам J
    for(j = 0;j < n;j++)
      {
       printf(" %2.0f  ",a[i][j]); // Вывод Aij
       if(a[i][j] < 0) k = k + 1;
      }
// Формирование элементов B(I)
    b[i] = a[i][i] - k;
    if(b[i] > 0) p = p*b[i]; // Вычисление P
   }
 printf("  \n \t ARRAY B(3): \n ");
 for(i = 0;i < n;i++)     // Вывод массива B
   printf(" %2.0f ",b[i]);
 printf(" \n   P =  %.0f ",p); // Вывод P
 getch();
 return;
}
```

6. Макет печати результатов

По C-программе



По Fortran-программе

```
Матрица A(3,3)
   1   -2   -3
   2    6   -4
  -1    3    2
Массив B(3)
  -1   5  1
    P =  5
```

Как видно, результаты ручного и машинного расчётов совпадают.

7. Листинг Fortran-программы

Программный код

```
! Формирование одномерного массива B
PROGRAM MACCIF
! Объявление типа данных – целого типа
INTEGER :: A(1:3,1:3),B(3)
INTEGER :: I,J,K,N = 3,P = 1
! Задание матрицы по столбцам
DATA A/1,2,-1,-2,6,3,-3,-4,2/
! Вывод матрицы по строкам
WRITE(*,*) '       МАТРИЦА A(3,3) '
WRITE(*,2)((A(I,J),J = 1,3),I = 1,3)
2 FORMAT(2X,3I4)
DO I = 1,N ! Внешний цикл по I (по строкам)
  K = 0
  DO J = 1,N ! Внутренний цикл по J (по столбцам)
    IF(A(I,J) < 0) K = K + 1
  END DO ! Конец цикла по J
! Вычисление элементов B(I)
B(I) = A(I,I) – K
IF(B(I) > 0) P = P * B(I)
END DO ! Конец цикла по I
! Вывод массива B и произведения P
WRITE(*,*) '     МАССИВ B(3) '
WRITE(*,"( 3I4/2X,A,I3)") B,' P = ',P
END
```

Результаты работы представлены в макете печати (п.6).

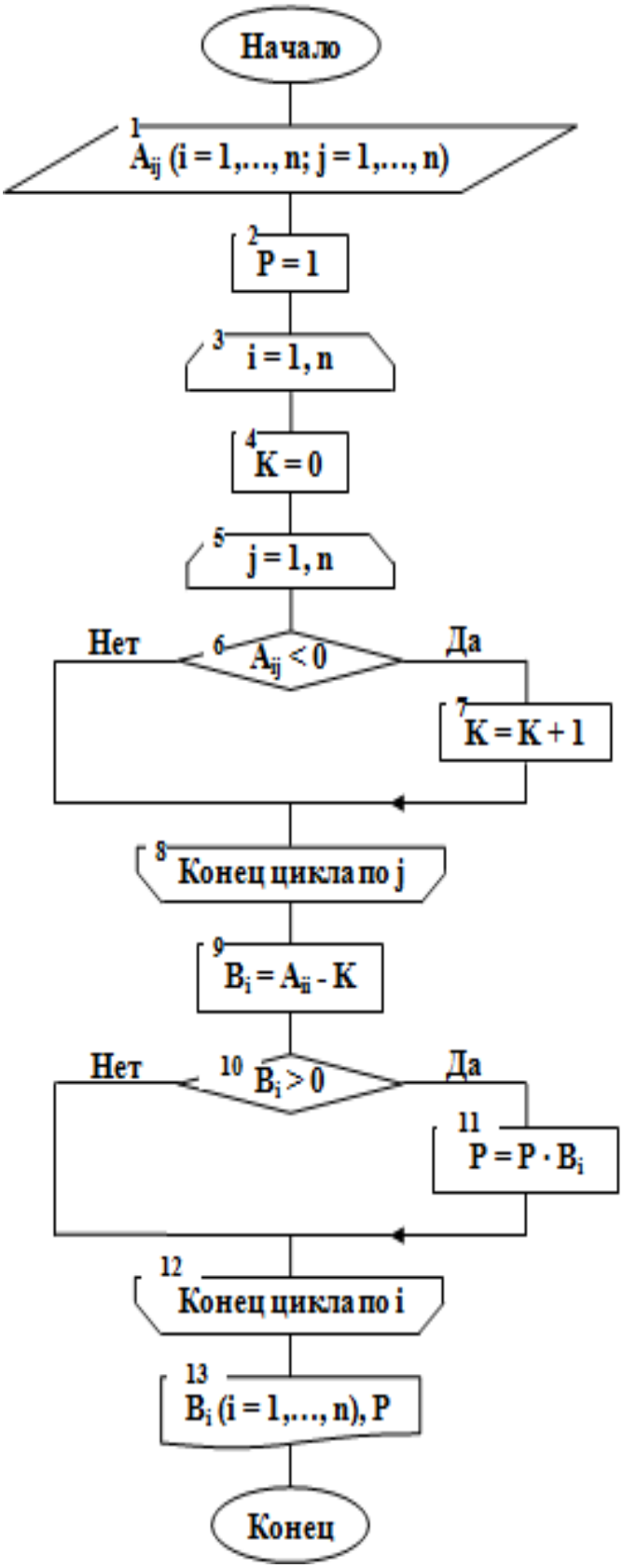


Рис. 2.30 Схема алгоритма к лабораторной работе 17 «Формирование одномерного массива»

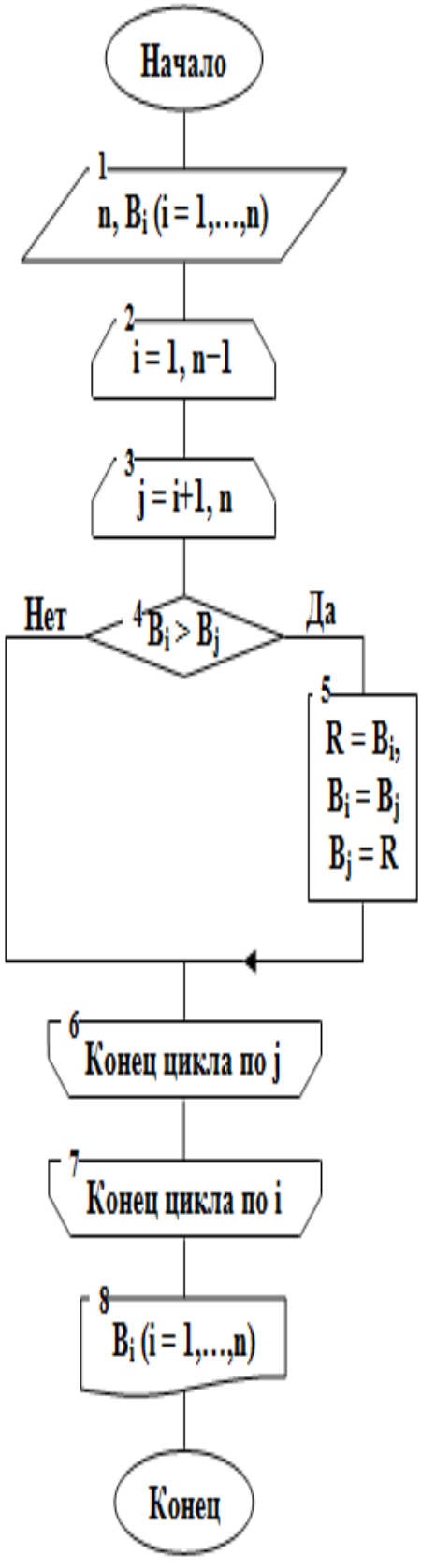


Рис. 2.31 Схема алгоритма к лабораторной работе 17 «Упорядочение массива»

8. Листинг Basic-программы «Формирование массива»

```
CLS
DIM A(3, 3), B(3)
' Задание матрицы A по строкам
DATA  1,-2,-3, 2, 6,-4,-1, 3, 2
PRINT TAB(10); " Матрица A(3,3)"
P = 1 : N = 3
FOR I = 1 TO N ' Внешний цикл по I (по строкам)
 K = 0
 FOR J = 1 TO N ' Внутренний цикл по J (по столбцам)
   READ A(I, J)
   PRINT USING " ### "; A(I, J);
   IF A(I, J) < 0 THEN K = K + 1
 NEXT J ' Конец цикла по J
 B(I) = A(I, I) - K
 IF B(I) > 0 THEN P = P * B(I)
 PRINT ' Пропуск строки
NEXT I ' Конец цикла по I
PRINT TAB(10); " Массив B(3)"
' Цикл вывода массива B
FOR I = 1 TO N
 PRINT USING " ## "; B(I);
NEXT I
PRINT ' Пропуск строки
PRINT USING "   P = ## "; P ' Вывод P
END
```

Результаты представлены в макете печати (п.6).

Контрольное задание для проверки знаний

Тема. Двумерные массивы (матрицы)

Дана матрица A (nxn), n = 6. В матрице можно выделить три совокупности элементов A_{ij}: главной диагонали (при **i = j**), верхнего и нижнего треугольника (при **i < j** и **i > j**). Обход верхнего треугольника удобнее осуществлять по строкам, а нижнего – по столбцам. В этом случае для верхнего треугольника параметр внешнего цикла, номер строки **i**, изменяется в интервале **[1; n – 1]**, а параметр внутреннего цикла, номер столбца **j**, – в интервале **[i+1;n]**. Для нижнего треугольника параметр внешнего цикла, номер столбца **j**, изменяется в интервале **[1; n – 1]**, а параметр внутреннего цикла, номер строки **i**, – в интервале **[j + 1; n]**. Матрица задаётся самостоятельно в соответствии с условием задачи.

Задания для лабораторной работы

Составить схему и программу для следующей задачи.

Определить:

№	Задача
1	Наименьший из положительных элементов чётных строк и определить его принадлежность верхнему, нижнему треугольнику, либо главной диагонали. Если это элемент верхнего треугольника, то найти сумму его отрицательных элементов.
2	Наибольший из отрицательных элементов нечётных столбцов и определить его принадлежность верхнему, нижнему треугольнику, либо главной диагонали. Если это элемент нижнего треугольника, то найти произведение его положительных элементов.
3	Номер столбца с наименьшим из положительных элементов верхнего треугольника. Если это чётный столбец, то найти сумму его элементов, если нечётный – их произведение.
4	Номер строки с наибольшим из отрицательных элементов нижнего треугольника. Если строка – чётная, то найти количество её положительных элементов, если нечётная – количество отрицательных элементов.
5	Номер строки и столбца наименьшего из элементов верхнего треугольника. Найти сумму положительных элементов строки и произведение отрицательных элементов столбца.
6	Номер строки и столбца наибольшего из элементов нижнего треугольника. Найти наименьший из положительных элементов строки и количество отрицательных элементов столбца.
7	Наименьший элемент матрицы и определить его принадлежность верхнему, нижнему треугольнику, либо главной диагонали. Если это элемент верхнего треугольника, то найти среднее арифметическое его отрицательных элементов.
8	Наибольший элемент матрицы и определить его принадлежность верхнему, нижнему треугольнику, либо главной диагонали. Если это элемент нижнего треугольника, то найти произведение его положительных элементов.
9	Сумму положительных элементов главной диагонали и наименьший из элементов верхнего треугольника, превышающих это значение.
10	Среднее арифметическое элементов нижнего треугольника, не превышающих значение наибольшего элемента главной диагонали.

3. Задания на выполнение лабораторных работ

Для таблиц 1-17 приведены примеры выполнения лабораторных работ (раздел 2).

Таблица 3.1 Вычисление значения выражения

Математические функции – табл. 5.1 (стр.136)
Пример выполнения лабораторной работы – стр.5-8

Задания для лабораторной работы 1

Составить программу вычисления величины Z .

№	Формула для вычисления Z	Исходные данные		Значение Z
		a	b	
1	$Z=\frac{a\sqrt{\ln 2a+\ln b}}{b+\sqrt{\ln ab+\ln 3}}-\frac{a}{b}\sqrt{\ln a+\ln b}$	2	1,5	– 0,5
2	$Z=\sqrt{\sin^2 8^\circ+\sin^2 a}+\frac{\sin^2\pi+\sin^2 b}{a(\sin^2 ab+\sin^2 2b)}$	– 2	1,25	0,2913
3	$Z=\frac{\ln^2 a+\ln^2 b}{a(\ln^2 3+\ln^2 3a)}-(\ln^2\frac{b}{a}+\ln^2 ab)^2$	1,5	2	– 1,5394
4	$Z=\sqrt{\lvert\operatorname{tg} a-\operatorname{tg} b\rvert}-\frac{b+(\operatorname{tg} 3^\circ-\operatorname{tg} ab)}{(\operatorname{tg} 2a-\operatorname{tg} 0{,}2\pi)}$	0,8	3	1,1962
5	$Z=\frac{(\cos 2a+\cos b)^2}{a(\cos 5^\circ+\cos 0{,}3\pi)}+\sqrt{\lvert\cos a+\cos 2b\rvert}$	1,2	1	0,2524
6	$Z=(\cos^2\frac{\pi}{3}+\ln a)^2-\frac{\cos^2 3^\circ+\ln\lvert b\rvert}{a(\cos^2 b+\ln 2)}$	3	– 2	1,1684
7	$Z=\frac{(\sin 2a+\cos b)^2}{b(\sin 3^\circ+\cos a)}+\sqrt{\lvert\sin b+\cos\frac{a}{b}\rvert}$	– 1	1,7	2,4201

8	$Z = \sqrt{\lvert \ln b + \sin 2^\circ \rvert} - \dfrac{(\ln 2 + \sin 0{,}3\pi)^2}{a(\ln a + \sin b)}$	2	3,1	– 0,4556
9	$Z = \dfrac{(\cos b - \operatorname{tg} a)^2}{\cos 6^\circ - \operatorname{tg} b} + \sqrt{\left\lvert \cos a - \operatorname{tg} \dfrac{a}{b} \right\rvert}$	2,4	– 2	1,1446
10	$Z = \sqrt{\lvert \cos a + \ln 3 \rvert} - \dfrac{(\cos \pi + \ln b)^2}{\cos 3^\circ + \ln \lvert a \rvert}$	– 3	1,5	0,16103
11	$Z = \dfrac{(\ln \lvert a \rvert - \operatorname{tg} b)^2}{\lvert \ln b - \operatorname{tg} 3^\circ \rvert} + \sqrt{\left\lvert \ln 2b - \operatorname{tg} \dfrac{\pi}{3} \right\rvert}$	– 2	1,85	31,7154
12	$Z = \sqrt{\lvert \cos^2 a - \sin^2 b \rvert} - \dfrac{(\cos^2 b - \sin^2 3^\circ)^2}{a(\cos^2 \pi - \sin^2 a)}$	1,2	2,35	– 0,9182
13	$Z = \dfrac{\sin^2 b + \ln a}{b(\sin^2 3^\circ + \ln 2)} + (\sin^2 a + \ln \lvert b \rvert)^2$	2	– 3	3,3657
14	$Z = \sqrt{\lvert e^{-b} + \operatorname{tg} 3^\circ \rvert} - \dfrac{e^{2a} + \operatorname{tg} b}{b - (e^b + \operatorname{tg} a)^2}$	1,4	0,5	1,1209
15	$Z = \dfrac{\ln b - e^{-a}}{\ln \lvert a \rvert - e^3} + (\ln 2b - e^b)^2$	– 2	1,5	11,8053
16	$Z = (\sin \dfrac{\pi}{3} + e^{2b})^2 - \dfrac{\sin 5^\circ + e^a}{\sin \lvert a \rvert + e^{-a}}$	– 2	1	68,1193
17	$Z = \dfrac{e^3 - \cos 4^\circ}{\sqrt{\lvert e^{-a} - \cos 2b \rvert}} + (e^b - \cos \dfrac{\pi}{3})^2$	1	2,7	243,73
18	$Z = (\ln \lvert a \rvert + \cos b)^2 - \dfrac{\ln 3 + \cos a}{b(\ln \sqrt{b} + \cos 3^\circ)}$	– 3	2	0,4254
19	$Z = \dfrac{\sin a + \ln^2 b}{a(\sin 3^\circ + \ln^2 b)} + (\sin \dfrac{\pi}{3} + \ln^2 3)^2$	1,5	2	6,1465
20	$Z = (\sqrt{\dfrac{a}{b}} + \sin 2a)^2 - \dfrac{\sqrt{a} + \sin 5^\circ}{b(\sqrt{b} + \sin a)}$	1	2,5	2,1975

Таблица 3.2 Вычисление значения величины, заданной в форме ветвления

Пример выполнения лабораторной работы – стр.8-11
Математические функции – табл. 5.1 (стр.136)
Составить схему и программу вычисления величины Z.

Для выполнения лабораторной работы необходимо:

1. При выполнении программы по запросу с экрана ввести с клавиатуры поочерёдно каждое из значений A, B, K. Результат Z сравнить с приведённым в таблице.
2. Действие п.1 повторить для каждой из ветвей алгоритма.
3. При несовпадении результата проанализировать программу, исправить ошибки и повторить действие п.1.

Задания для лабораторной работы 2

№	Формула для вычисления Z	Исходные данные			Значение Z
		a	b	k	
1	$Z=\begin{cases} k\cdot\max(a,b) & \text{при } a\ge 1 \text{ и } b>0 \\ b^2-a\cdot k & \text{при } a\ge 1 \text{ и } b\le 0 \\ a+b\cdot k & \text{при } a<1 \text{ и } b=1 \\ b+a\cdot k^2 & \text{при } a<1 \text{ и } b\ne 1 \end{cases}$	3 3 –1 –3	1 –1 1 2	2 1 2 2	6 –2 1 –10
2	$Z=\begin{cases} k\cdot\min(a,b) & \text{при } b>0 \text{ и } a=1 \\ a\cdot b+2\cdot k & \text{при } b>0 \text{ и } a\ne 1 \\ a\cdot k^2+b & \text{при } b\le 0 \text{ и } a>b \\ a+b\cdot k & \text{при } b\le 0 \text{ и } a\le b \end{cases}$	1 3 1 –2	5 2 –3 –1	4 2 2 3	4 10 1 –5
3	$Z=\begin{cases} b\cdot\max(a,k) & \text{при } a<0 \text{ и } b>1 \\ b-a\cdot k & \text{при } a<0 \text{ и } b\le 1 \\ a+b\cdot k^2 & \text{при } a\ge 0 \text{ и } a>b \\ k\cdot b-a & \text{при } a\ge 0 \text{ и } a\le b \end{cases}$	–1 –4 2 1	2 –1 1 2	3 1 2 2	6 3 6 3
4	$Z=\begin{cases} a\cdot\min(k,b) & \text{при } a>b \text{ и } b=1 \\ k-a\cdot b & \text{при } a>b \text{ и } b\ne 1 \\ a\cdot b^2+k & \text{при } a\le b \text{ и } a>0 \\ a\cdot b+k^2 & \text{при } a\le b \text{ и } a\le 0 \end{cases}$	3 3 1 –5	1 2 2 –4	3 4 2 2	3 –2 6 24

№	Формула для вычисления Z	Исходные данные			Значение Z
		a	b	k	
5	$Z=\begin{cases} a+\max(b,k) & \text{при } a=b \text{ и } a>0 \\ a+k\cdot b^2 & \text{при } a=b \text{ и } a\le 0 \\ a\cdot b+2k & \text{при } a\ne b \text{ и } b>1 \\ a^2+b\cdot k & \text{при } a\ne b \text{ и } b\le 1 \end{cases}$	2 –1 2 2	2 –1 3 –1	3 2 2 2	5 1 10 2
6	$Z=\begin{cases} b-\min(k,a) & \text{при } a=1 \text{ и } b\ge 0 \\ b+a\cdot k^3 & \text{при } a=1 \text{ и } b<0 \\ a+b\cdot k & \text{при } a\ne 1 \text{ и } a>b \\ a\cdot b-k & \text{при } a\ne 1 \text{ и } a\le b \end{cases}$	1 1 3 2	4 –1 2 3	3 2 1 2	3 7 5 4
7	$Z=\begin{cases} k+\max(a,b) & \text{при } b\ne 1 \text{ и } a<0 \\ a^2\cdot b-2k & \text{при } b\ne 1 \text{ и } a\ge 0 \\ a\cdot b^2+k & \text{при } b=1 \text{ и } a\le b \\ a\cdot b+k^2 & \text{при } b=1 \text{ и } a>b \end{cases}$	–2 3 –1 2	2 3 1 1	3 2 2 2	5 5 1 6
8	$Z=\begin{cases} b\cdot\min(2a,k) & \text{при } b>1 \text{ и } a>0 \\ a\cdot b+k & \text{при } b>1 \text{ и } a\le 0 \\ a\cdot k^2+b & \text{при } b\le 1 \text{ и } a\ne 1 \\ b-k\cdot a^2 & \text{при } b\le 1 \text{ и } a=1 \end{cases}$	3 –1 –2 1	2 2 –1 –5	4 1 2 2	8 –1 –9 –7
9	$Z=\begin{cases} a\cdot\max(k,2b) & \text{при } a\le 0 \text{ и } b=1 \\ a+b\cdot k & \text{при } a\le 0 \text{ и } b\ne 1 \\ a\cdot b+2k^2 & \text{при } a>0 \text{ и } a>b \\ a\cdot k-b & \text{при } a>0 \text{ и } a\le b \end{cases}$	–2 –3 3 1	1 2 1 3	3 1 2 2	–6 –1 11 –1
10	$Z=\begin{cases} k\cdot\min(b,2a) & \text{при } b\le 0 \text{ и } a\ge 0 \\ 2k-a\cdot b & \text{при } b\le 0 \text{ и } a<0 \\ a+b\cdot k^2 & \text{при } b>0 \text{ и } a>1 \\ k-a\cdot b^2 & \text{при } b>0 \text{ и } a\le 1 \end{cases}$	1 –1 2 –1	–4 –2 1 2	2 2 2 3	–8 2 6 7
11	$Z=\begin{cases} k\cdot\max(a,2b) & \text{при } b>1 \text{ и } a>b \\ 2a+b\cdot k & \text{при } b>1 \text{ и } a\le b \\ a\cdot b+k^2 & \text{при } b\le 1 \text{ и } a<0 \\ a\cdot b+2k & \text{при } b\le 1 \text{ и } a\ge 0 \end{cases}$	3 1 –9 2	2 2 –1 –5	3 1 2 2	12 4 13 –6
12	$Z=\begin{cases} a\cdot\min(b,k^2) & \text{при } a<b \text{ и } a\le 0 \\ a\cdot b+2k & \text{при } a<b \text{ и } a>0 \\ a\cdot k+b^3 & \text{при } a\ge b \text{ и } b=1 \\ k^2+a\cdot b & \text{при } a\ge b \text{ и } b\ne 1 \end{cases}$	–1 1 2 2	9 3 1 –1	2 2 3 1	–4 7 7 –1

№	Формула для вычисления Z	Исходные данные			Значение Z
		a	b	k	
13	$Z=\begin{cases} b\cdot\max(a,k^2) & \text{при } a>0 \text{ и } a\le b \\ a^2-b\cdot k & \text{при } a>0 \text{ и } a>b \\ b-a\cdot k^2 & \text{при } a\le 0 \text{ и } b<1 \\ a\cdot b+3k & \text{при } a\le 0 \text{ и } b\ge 1 \end{cases}$	2 4 –2 –4	3 1 –1 1	2 3 2 2	12 13 7 2
14	$Z=\begin{cases} k\cdot\min(a,b^2) & \text{при } a<1 \text{ и } b\le 0 \\ a\cdot b-3k & \text{при } a<1 \text{ и } b>0 \\ b+a\cdot k^2 & \text{при } a\ge 1 \text{ и } a<b \\ b^2-a\cdot k & \text{при } a\ge 1 \text{ и } a\ge b \end{cases}$	–1 –1 1 1	–4 2 2 –1	3 1 1 2	–3 –5 3 –1
15	$Z=\begin{cases} k\cdot\max(b,a^2) & \text{при } b=1 \text{ и } a\ge 0 \\ a\cdot b+2k & \text{при } b=1 \text{ и } a<0 \\ b\cdot k^2-a & \text{при } b\ne 1 \text{ и } a=b \\ k+a\cdot b & \text{при } b\ne 1 \text{ и } a\ne b \end{cases}$	3 –2 2 3	1 1 2 2	2 2 2 4	18 2 6 10
16	$Z=\begin{cases} k\cdot\min(b,a^2) & \text{при } a\ge 0 \text{ и } b<1 \\ a\cdot b^2+2k & \text{при } a\ge 0 \text{ и } b\ge 1 \\ a\cdot k+b & \text{при } a<0 \text{ и } b=1 \\ a+b\cdot k^2 & \text{при } a<0 \text{ и } b\ne 1 \end{cases}$	5 1 –1 –3	–4 2 1 2	2 2 2 2	–8 8 –1 5
17	$Z=\begin{cases} k\cdot\max(a,b^3) & \text{при } a=b \text{ и } a>1 \\ a\cdot k^2+b & \text{при } a=b \text{ и } a\le 1 \\ a\cdot k-b^2 & \text{при } a\ne b \text{ и } a>0 \\ b\cdot k-a^3 & \text{при } a\ne b \text{ и } a\le 0 \end{cases}$	2 1 2 –1	2 1 3 4	1 2 2 3	8 5 –5 13
18	$Z=\begin{cases} b+\min(a,k) & \text{при } b>0 \text{ и } a=b \\ b+a\cdot k^2 & \text{при } b>0 \text{ и } a\ne b \\ a^3-k\cdot b & \text{при } b\le 0 \text{ и } a\ge 1 \\ k+a\cdot b^2 & \text{при } b\le 0 \text{ и } a<1 \end{cases}$	2 1 1 –1	2 2 –1 –2	4 2 2 5	4 6 3 1
19	$Z=\begin{cases} a\cdot\max(b,k) & \text{при } b<1 \text{ и } a=1 \\ b^2-a\cdot k & \text{при } b<1 \text{ и } a\ne 1 \\ b\cdot k-a^3 & \text{при } b\ge 1 \text{ и } a<b \\ a\cdot b-k^2 & \text{при } b\ge 1 \text{ и } a\ge b \end{cases}$	1 –1 –2 2	–4 –2 1 1	3 2 2 2	3 6 10 –2
20	$Z=\begin{cases} k\cdot\min(a^3,b) & \text{при } a>1 \text{ и } b\le 0 \\ 2\cdot k-a\cdot b & \text{при } a>1 \text{ и } b>0 \\ a\cdot k+b^2 & \text{при } a\le 1 \text{ и } a\ne b \\ b\cdot k-a^3 & \text{при } a\le 1 \text{ и } a=b \end{cases}$	5 4 1 1	–4 2 3 1	1 3 2 2	–4 –2 11 1

Таблица 3.3 Вычисление значения величины, заданной в форме ветвления

Пример выполнения лабораторной работы – стр.11-14
Математические функции – табл. 5.1 (стр.136)
Составить схему и программу вычисления величины Z.

Для выполнения лабораторной работы необходимо:

1. При выполнении программы по запросу с экрана ввести с клавиатуры поочерёдно каждое из значений A, B, C. Результат Z сравнить с приведённым в таблице.
2. Действие п.1 повторить для каждой из ветвей алгоритма.
3. При несовпадении результата проанализировать программу, исправить ошибки и повторить действие п.1.

Задания для лабораторной работы 3

№	Формула для вычисления Z	Исходные данные			Значение Z
		a	b	c	
1	$Z=\begin{cases}\max\{b, a\cdot\min(a^2, 2c)\} & \text{при } c\le a\\ b^2+\min(c, 2a) & \text{при } c>a \text{ и } b=1\\ \min(2a, \sqrt{\lvert b\rvert})-c & \text{при } c>a \text{ и } b\ne 1\end{cases}$	2	–1	1	4
		1	1	3	3
		–1	4	3	1
2	$Z=\begin{cases}\max(a, b^2, \sqrt{\lvert c\rvert}) & \text{при } a<1\\ c+\min(b, a^2) & \text{при } a\ge 1 \text{ и } b>c\\ \min(2b, c^2)-a & \text{при } a\ge 1 \text{ и } b\le c\end{cases}$	–2	3	9	9
		2	5	3	7
		2	3	4	4
3	$Z=\begin{cases}\min\{a, b\cdot\max(b^2, c)\} & \text{при } a\ge b\\ b+\max(a, 2c) & \text{при } a<b \text{ и } c>0\\ \max(b, \sqrt{\lvert c\rvert})-a & \text{при } a<b \text{ и } c\le 0\end{cases}$	3	2	5	3
		2	3	4	11
		2	3	–4	1
4	$Z=\begin{cases}\min(b^2, \sqrt{\lvert a\rvert}, 2c) & \text{при } b>c\\ a-\max(2b, c) & \text{при } b\le c \text{ и } a>1\\ \max(a^2, c)+b & \text{при } b\le c \text{ и } a\le 1\end{cases}$	4	3	2	2
		6	2	3	2
		–2	2	3	6

№	Формула для вычисления Z	Исходные данные			Значение Z
		a	b	c	
5	$Z=\begin{cases} b+\min(2a,\sqrt{\lvert c\rvert}) & \text{при } a=b \\ \min(b^2,c)-a & \text{при } a\neq b \text{ и } c\le 0 \\ \max(b,a^2,2c) & \text{при } a\neq b \text{ и } c>0 \end{cases}$	3	3	16	7
		3	2	–1	–4
		3	2	3	9
6	$Z=\begin{cases} \max(2b,c^3)+a^2 & \text{при } a>c \\ 2a-\min(c^2,b) & \text{при } a\le c \text{ и } b=1 \\ \min\{a,b\cdot\min(b^3,c)\} & \text{при } a\le c \text{ и } b\neq 1 \end{cases}$	3	1	2	13
		2	1	4	3
		2	3	9	2
7	$Z=\begin{cases} b+\max(a^2,c) & \text{при } a\ge b \\ \min(\sqrt{\lvert b\rvert},2a,c^2) & \text{при } a<b \text{ и } c>a \\ c-\max(2b,a^3) & \text{при } a<b \text{ и } c\le a \end{cases}$	2	1	5	6
		–1	4	3	–2
		1	4	–2	–10
8	$Z=\begin{cases} \min\{b,a\cdot\max(c,2a)\} & \text{при } b\ge 0 \\ c^2-\max(a,b) & \text{при } b<0 \text{ и } a\neq c \\ \max(2c,b^2)+a^2 & \text{при } b<0 \text{ и } a=c \end{cases}$	3	2	3	2
		1	–1	2	3
		3	–1	3	15
9	$Z=\begin{cases} \min\{c,b\cdot\min(2a,b^3)\} & \text{при } b>c \\ a-\min(\sqrt{\lvert b\rvert},c^2) & \text{при } b\le c \text{ и } a>1 \\ \max(a^3,3b)+c & \text{при } b\le c \text{ и } a\le 1 \end{cases}$	2	3	1	1
		3	4	5	1
		–1	3	6	12
10	$Z=\begin{cases} \min(b^2,2a)-ac & \text{при } a\le 1 \\ \max(2b,a^2,c^2) & \text{при } a>1 \text{ и } b\neq c \\ b^2+\min(a,3c) & \text{при } a>1 \text{ и } b=c \end{cases}$	–2	2	3	2
		2	3	4	16
		2	3	3	11
11	$Z=\begin{cases} \min\{a^2,c\cdot\max(c^3,\sqrt{\lvert b\rvert})\} & \text{при } b\ge 0 \\ \max(a,b^3)+2c & \text{при } b<0 \text{ и } a\ge c \\ a-\max(b,c^2) & \text{при } b<0 \text{ и } a<c \end{cases}$	3	4	2	9
		4	–1	3	10
		1	–1	2	–3

№	Формула для вычисления Z	Исходные данные			Значение Z
		a	b	c	
12	$Z=\begin{cases}\min(a^3,\sqrt{\lvert c\rvert},b^2) & \text{при } a>0\\ c+\max(b,2a) & \text{при } a\le 0 \text{ и } b=c\\ \max(2b,c^2)+a & \text{при } a\le 0 \text{ и } b\ne c\end{cases}$	2 –1 –1	3 3 2	4 3 3	2 6 8
13	$Z=\begin{cases}\max\{b^2,c\cdot\min(a,2c)\} & \text{при } a>c\\ a+\min(b,c^2) & \text{при } a\le c \text{ и } a>b\\ \min(2a,\sqrt{\lvert b\rvert})-c^2 & \text{при } a\le c \text{ и } a\le b\end{cases}$	3 2 2	4 –1 4	2 3 3	16 1 –7
14	$Z=\begin{cases}\max(a,b,c^2) & \text{при } c\ge 1\\ \min(2c,\sqrt{\lvert a\rvert})-b & \text{при } c<1 \text{ и } b=a\\ a+\min(c^2,2b) & \text{при } c<1 \text{ и } b\ne a\end{cases}$	4 4 4	1 4 3	3 –1 –1	9 –6 5
15	$Z=\begin{cases}\min\{a,b\cdot\max(c,b)\} & \text{при } a>c\\ \max(c,b^3)+a & \text{при } a\le c \text{ и } b>1\\ \sqrt{\lvert c\rvert}-\max(2a,b^2)+b^2 & \text{при } a\le c \text{ и } b\le 1\end{cases}$	5 –1 1	2 2 –1	4 1 9	5 7 1
16	$Z=\begin{cases}\max(c^2\sqrt{\lvert a\rvert})-b & \text{при } a>b\\ \lvert c\rvert+\max(2a,b^3) & \text{при } a\le b \text{ и } c=1\\ \min(b^2,2c,a^3) & \text{при } a\le b \text{ и } c\ne 1\end{cases}$	4 2 2	2 3 3	3 1 4	7 28 4
17	$Z=\begin{cases}2c+\min(a^2,b) & \text{при } a<b\\ \min(2b,c)-2a & \text{при } a\ge b \text{ и } c\ge 0\\ \max(2a,b^2,\sqrt{\lvert c\rvert}) & \text{при } a\ge b \text{ и } c<0\end{cases}$	–1 3 3	3 1 2	1 4 –4	3 –4 6
18	$Z=\begin{cases}\max\{b,c\cdot\min(a,b)\} & \text{при } b=c\\ b-\min(a^2,c) & \text{при } b\ne c \text{ и } a\ge 1\\ \min(a,2b)+c^2 & \text{при } b\ne c \text{ и } a<1\end{cases}$	3 4 –1	2 2 4	2 3 3	4 –1 8

№	Формула для вычисления Z	Исходные данные			Значение Z
		a	b	c	
19	$Z=\begin{cases}\min(a,2c,b^2) & \text{при } a=c\\ b+\max(2c,a^3)-b & \text{при } a\neq c \text{ и } b\geq 0\\ \max(b^2,2a)-c & \text{при } a\neq c \text{ и } b<0\end{cases}$	2	3	2	2
		2	4	3	12
		2	–1	3	1
20	$Z=\begin{cases}\min\left\{\sqrt{\lvert b\rvert}, c\cdot\max(c^2,a)\right\} & \text{при } b\geq 1\\ \max(a,b^2)+c & \text{при } b<1 \text{ и } a\geq c\\ a-\max(2c,b) & \text{при } b<1 \text{ и } a<c\end{cases}$	2	9	1	2
		3	–2	2	6
		2	–2	3	–4

Таблица 3.4 Исследование функции одной переменной

Пример выполнения лабораторной работы – стр.17-22

Математические функции – табл. 5.1 (стр.136)

Порядок выполнения лабораторной работы – раздел 1 (стр.4)

Составить схему и программу решения для следующей задачи.

Вычислить таблицу значений аргумента **X** и функции **Y = F(X)** при условии, что аргумент **X** изменяется в интервале $\mathbf{X_{min} \le X \le X_{max}}$ с шагом $\Delta\mathbf{X}$.

При решении задач используются алгоритмы вычисления суммы, произведения, количества, наибольшего (наименьшего) значения.

Задания для лабораторной работы 4

Протабулировать функцию и найти:

№	Задача	Исходные данные		
		X_{min}	X_{max}	ΔX
1	Сумму положительных значений функции $y = \cos^3(x) - e^{-x}$	0	1	0,1
2	Произведение отрицательных значений функции $y = \ln^2(x\cdot\sqrt{x}) - e^{-x}$	0,2	2	0,2
3	Количество положительных значений функции $y = \ln^3(1+\sqrt{\lvert x\rvert}) - e^{-x}$	0,4	2,2	0,2
4	Сумму и произведение значений функции. Какая из этих величин больше? $y = \sin^2(2\cdot x) - e^{x}$	–1	0,6	0,2

№	Задача	Исходные данные		
		X_{min}	X_{max}	ΔX
5	Значение аргумента x, при котором достигается наименьшее значение функции $y = \sqrt{\lvert x\rvert} - \sin^3(2 \cdot x)$	1	2	0,1
6	Наименьшее из положительных значений функции $y = \sin^3 x - e^{-x}$	0,3	1,2	0,1
7	Произведение и количество положительных значений функции $y = tg^3(\sqrt{\lvert x\rvert}) - e^{-2x}$	0,1	1	0,1
8	Сумму положительных и произведение отрицательных значений функции $y = tg^3(3 \cdot x) - \sqrt{\lvert x\rvert}$	0,1	1	0,1
9	Какому значению аргумента x соответствует наибольшее значение функции $y = \cos^2 x + \sqrt{\lvert x \cdot \ln(x)\rvert}$	0,1	1	0,1
10	Произведение положительных и произведение отрицательных значений функции. $y = tg^3(2 \cdot x) - \ln\lvert x\rvert$	0,2	2	0,2
11	Произведение значений функции. Вывести сообщение о том, положительна эта величина или отрицательна. $y = \sqrt{\lvert x\rvert} - tg^2(3 \cdot x)$	1,5	2,3	0,1
12	Категория каких значений функции преобладает на интервале, положительных или отрицательных? $y = \sqrt{\lvert x\rvert} + x \cdot tg^3 x$	1	2	0,1
13	Сумму и количество положительных значений функции $y = \sin^2(2 \cdot x) - e^{-x}$	0	1,6	0,2
14	Наименьшее значение функции. Определить знак этой величины. $y = \sin^2(3 \cdot x) + \sqrt{\lvert 1 - x^2\rvert}$	1	2,6	0,2

№	Задача	Исходные данные		
		X_{min}	X_{max}	ΔX
15	Сумму положительных и количество отрицательных значений функции $y=\cos^2(x)-e^{-x}$	–1	1	0,2
16	Сумму и произведение положительных значений функции $y=\sin^2(3\cdot x)-e^{-x}$	0	1,6	0,2
17	Среднее арифметическое отрицательных значений функции $y = tg^{3}(2\cdot x)-\sqrt{\lvert x\rvert}$	0,1	1	0,1
18	Произведение отрицательных и количество положительных значений функции $y = tg^{2}(3\cdot x)-e^{-x}$	0,2	1,8	0,2
19	Сумму положительных и сумму отрицательных значений функции $y = \cos^{3}(2\cdot x)-\sqrt{\lvert x\rvert}$	0	0,9	0,1
20	Среднее арифметическое значение функции. Вывести сообщение о том, положительна эта величина или отрицательна. $y=tg^2(3\cdot x)-e^{-2\cdot x}$	0	1	0,1

Таблица 3.5 Исследование функции одной переменной, зависящей от параметра

Пример выполнения лабораторной работы – стр.22-27

Математические функции – табл. 5.1 (стр.136)

Порядок выполнения лабораторной работы – раздел 1 (стр.4)

Составить схему и программу решения для следующей задачи.

Вычислить таблицу значений функции $y = f(X,\alpha)$ и аргумента X на интервале [X_{min}, X_{max}] с шагом ΔX при условии, что параметр α принимает ряд фиксированных значений α_1 и α_2. Таким образом, вычисление функции можно представить в виде:

$$y = \begin{cases} f(X,\alpha_1), & \text{если} \quad X_{\min} \le X \le X_{\text{ср}} \\ f(X,\alpha_2), & \text{если} \quad X_{\text{ср}} < X \le X_{\max}, \end{cases}$$

где $X_{ср}$ – значение аргумента X из интервала $X_{min} < X_{ср} < X_{max}$.

При решении используются алгоритмы вычисления суммы, произведения, количества, наибольшего (наименьшего) значения.

Задания для лабораторной работы 5

Протабулировать функцию и найти:

№	Задача	Исходные данные					
		X_{min}	X_{cp}	X_{max}	ΔX	α_1	α_2
1	Наименьшее значение функции и соответствующее ему значение аргумента X $y = \sqrt{\lvert\alpha \cdot X\rvert} - tg^2 \alpha X + \frac{\alpha X}{1+\alpha}$	0	0,8	1,5	0,1	0,5	1
2	Наибольшее и наименьшее значение функции $y = \alpha X - \sqrt{\alpha + X^2} - \frac{\cos^2 X}{\alpha X^3}$	1	1,5	2	0,1	1	0,5
3	Наименьшее из положительных значений функции $y = \sqrt{\alpha + X^2} - \mathrm{tg}^2(\pi + \alpha X^2)$	0	1	2	0,2	1	2
4	Наибольшее из отрицательных значений функции $y = \alpha X + tg\alpha X - \alpha\sqrt{\alpha^2 + X^2}$	2	2,5	3	0,1	0,7	1,2
5	Наименьшее значение функции. Какого знака эта величина? $y = \alpha e^X - \sin^2 X\alpha - \alpha X^3$	0	1	2	0,25	0,5	1
6	Сумму отрицательных значений функции $y = \mathrm{tg}\frac{X}{\alpha} - \cos^2 \alpha X$	0,5	1	2	0,1	1	2
7	Произведение положительных и произведение отрицательных значений функции $y = \sin(X + \alpha) - e^{\alpha X} + \alpha X^2$	– 1	0	1	0,2	1	2
8	Сумму положительных и произведение отрицательных значений функции $y = \alpha X^3 - 1{,}5X + \alpha \mathrm{tg}\alpha X$	– 1	0	1	0,2	1	3
9	Сумму и произведение значений функции. Какая из этих величин больше? $y = X^\alpha - \alpha \mathrm{tg}\alpha X$	1,2	1,8	2,5	0,1	1	2

№	Задача	Исходные данные					
		X_{min}	$X_{ср}$	X_{max}	ΔX	α_1	α_2
10	Значения аргумента X, при которых достигаются наибольшее и наименьшее значения функции $y = \alpha X - 0{,}2\sin^2(\alpha + X)$	0	0,5	1	0,1	1	– 1
11	Произведение и количество отрицательных значений функции $y = \cos\alpha X - \alpha \cdot \mathrm{tg}^2 X + \alpha X$	– 1	– 0,5	0	0,1	1	2
12	Количество положительных и количество отрицательных значений функции. Каких на интервале больше? $y = \alpha X^3 + \sqrt{\lvert X\alpha \rvert} - \cos^2 \alpha X$	0	0,4	0,7	0,05	2	1
13	Сумму и количество положительных значений функции $y = \sin^2 \alpha X - \alpha \cdot \mathrm{tg}\, X$	0	1	2	0,2	1	0,5
14	Сумму положительных и сумму всех значений функции $y = X\alpha^2 - \mathrm{tg}\,\alpha X + \cos^2 X\alpha$	1	2	3	0,2	1	0,6
15	Среднее арифметическое отрицательных значений функции $y = X\alpha^2 - \cos^2 \alpha X$	0	0,5	1	0,1	1	2
16	Сумму и произведение положительных значений функции $y = tg^2 \alpha X - \sqrt{\lvert \alpha^2 - \ln X \rvert}$	1	2	2,5	0,1	1	2
17	Отношение произведения положительных значений функции к их количеству $y = 3X\alpha - 15\ln\lvert \alpha X - 3 \rvert - 40$	2,8	2,9	3	0,02	1	1,2
18	Сумму и количество отрицательных значений функции $y = \alpha X^3 + \sqrt{\lvert 10\alpha X \rvert} - \alpha e^X$	0	0,2	0,3	0,02	1	1,6
19	Сумму положительных и сумму отрицательных значений функции $y = \alpha e^X + \mathrm{tg}\,\alpha X - (X - \alpha)^2$	1,5	1,7	2	0,05	1	– 1
20	Произведение отрицательных значений функции $y = \sqrt{\lvert \alpha X \rvert} + \alpha X^3 + e^{\alpha X} - 60$	3	3,4	3,7	0,05	1	0,9

Таблица 3.6 Исследование функции одной переменной с различным шагом варьирования аргумента

Пример выполнения лабораторной работы – стр.28-33
Математические функции – табл. 5.1 (стр.136)
Порядок выполнения лабораторной работы – раздел 1 (стр.4)

Составить схему и программу решения для следующей задачи.

Вычислить таблицу значений функции Y = F(X) и аргумента X на интервале $[X_{min}, X_{max}]$ с шагом ΔX, причём шаг ΔX принимает ряд фиксированных значений ΔX_1 и ΔX_2 в зависимости от текущего значения аргумента X, а именно:

$$\Delta X = \begin{cases} \Delta X1, & \text{если } Xmin \le X \le Xcp \\ \Delta X2, & \text{если } Xcp < X \le Xmax \end{cases}$$

где X_{cp} – значение аргумента X на интервале $X_{min} < X_{cp} < X_{max}$.

При решении используются типовые алгоритмы вычисления суммы, произведения, наибольшего (наименьшего) значения.

Задания для лабораторной работы 6

Протабулировать функцию и найти:

№	Задача	Исходные данные				
		X_{min}	X_{cp}	X_{max}	ΔX_1	ΔX_2
1	Наименьшее значение функции. Какого знака эта величина? $y = \sin(1 - \sqrt{\lvert x \rvert}) + e^x$	0	1	2,5	0,2	0,25
2	Наибольшее значение функции $y = \sin^2 2x - \sqrt{\lvert x - \mathrm{tg}\, x \rvert}$	1	2	5	0,25	0,5
3	Сумму и количество отрицательных значений функции $y = \mathrm{tg}^2 x - \sqrt{\lvert x - \cos x \rvert}$	0	1	3,5	0,25	0,5
4	Количество положительных и количество отрицательных значений функции $y = \sin 2x + \mathrm{tg}^2 \sqrt{\lvert x \rvert}$	5,2	5,7	6,1	0,1	0,05

№	Задача	Исходные данные				
		X_{min}	$X_{ср}$	X_{max}	ΔX_1	ΔX_2
5	Наибольшее из отрицательных значений функции $y = \cos 2x + \mathrm{tg}^2 \ln\lvert x\rvert$	0,5	1	2,5	0,1	0,25
6	Наименьшее из положительных значений функции $y = \sqrt{\lvert x - \sin^2 x\rvert} - \mathrm{tg}(x + \frac{\pi}{3})$	3	4	7	0,2	0,5
7	Сумму и произведение значений функции. Какая из этих величин больше? $y = e^{-x} + \cos x$	3	7	9	0,5	0,25
8	Произведение положительных и количество отрицательных значений функции $y = \sin^2 3x - e^{-x}$	0	1	2	0,25	0,2
9	Среднее арифметическое значение функции. Какого знака эта величина? $y = \mathrm{tg}^2 x - 4 \cdot \mathrm{tg} x - 3$	1	2	5	0,25	0,5
10	Сумму положительных и количество отрицательных значений функции $y = \sin^2 x - \cos^2 x - \cos 2x$	0	1	4	0,2	0,5
11	Количество отрицательных значений функции $y = \mathrm{tg} x - \sin^2 2x$	3,5	4,5	7	0,25	0,5
12	Произведение отрицательных и сумму всех значений функции $y = \sqrt{\lvert x - \mathrm{tg} x\rvert} - \cos 2x$	–1	0	1,2	0,25	0,2
13	Среднее арифметическое положительных значений функции $y = e^x - \mathrm{tg}\lvert x\rvert$	–1	0,5	2,1	0,25	0,2
14	Произведение и количество отрицательных значений функции $y = x^2 + \mathrm{tg} x - \cos^3 x$	0	1,5	2	0,25	0,1
15	Сумму положительных и произведение отрицательных значений функции $y = \sqrt{\lvert x - \sin x\rvert} - \frac{1}{\mathrm{tg}^2 x}$	1	1,5	3	0,1	0,25

№	Задача	Исходные данные				
		X_{min}	X_{cp}	X_{max}	ΔX_1	ΔX_2
16	Сумму положительных и сумму всех значений функции $y = x^3 - \mathrm{tg}^2 x - \sin x$	1	4	5	0,5	0,2
17	Отношение произведения положительных значений функции к их количеству $y = x^3 - \sin^2 x + \mathrm{tg}(x + \frac{\pi}{3})$	0	1	1,5	0,2	0,1
18	Произведение положительных и произведение отрицательных значений функции $y = \mathrm{tg}^2 x - 3\cos x$	5	6	8,5	0,2	0,5
19	Сумму и произведение положительных значений функции $y = x + \mathrm{tg}\lvert x\rvert - \cos^2 x$	1,5	1,7	2,3	0,05	0,1
20	Произведение положительных и произведение всех значений функции $y = x - \mathrm{tg}2x + \sin^2 x$	0,25	1,5	2,4	0,25	0,1

Таблица 3.7 Исследование функции двух переменных

Пример выполнения лабораторной работы – стр.34-38

Математические функции – табл. 5.1 (стр.136)

Порядок выполнения лабораторной работы – раздел 1 (стр.4)

Составить схему и программу решения для следующей задачи.

Вычислить и напечатать таблицу значений функции Z = f(X,Y) и аргументов X,Y в заданной области при условии, что:

- аргумент X изменяется в интервале $X_0 \le X \le X_k$ с шагом ΔX;
- аргумент Y изменяется в интервале $Y_0 \le Y \le Y_k$ с шагом ΔY;

При решении используются алгоритмы вычисления суммы, произведения, количества, наибольшего (наименьшего) значения.

Задания для лабораторной работы 7

Протабулировать функцию и найти:

№	Задача	Исходные данные					
		X_0	X_k	ΔX	Y_0	Y_k	ΔY
1	Среднее арифметическое положительных значений функции $\mathbf{Z = x \cdot \ln^2(|y| + x^2) - e^{x \cdot y}}$	1	1,2	0,1	–2	1	1
2	Наименьшее из положительных значений функции $\mathbf{Z = x \cdot \cos^2 y - y}\sqrt{|x^2 - \sin y|}$	0,8	1,2	0,2	0,5	2	0,5
3	Наибольшее из отрицательных значений функции $\mathbf{Z = y \cdot \sin^2 x - x}\sqrt{|y^2 - \cos x|}$	1	1,2	0,1	0,5	2	0,5
4	Произведение и количество положительных значений функции $\mathbf{Z =}\sqrt{|\ln x - \cos y|} \mathbf{- y \cdot tg^2 x}$	1	3	1	0	1,5	0,5
5	Сумму и количество отрицательных значений функции $\mathbf{Z = y \cdot tg^2 x -}\sqrt{|\cos x|}\ \mathbf{e^{-y}}$	0,1	0,5	0,2	–3	3	2
6	Сумму положительных и сумму всех значений функции $\mathbf{Z = x \cdot tg^2 y -}\sqrt{|\cos y|}\ \mathbf{e^{x}}$	0,5	1,5	0,5	–1	2	1
7	Произведение отрицательных и количество положительных значений функции $\mathbf{Z =}\sqrt{|y - \sin^2 x|} \mathbf{- y \cdot e^{2x}}$	0,4	0,6	0,1	1	2	0,5
8	Сумму положительных значений функции $\mathbf{Z = x \cdot tg^2 y - y e^{-x}}$	0,1	0,3	0,1	0,5	1,5	0,5
9	Количество отрицательных значений функции $\mathbf{Z = tg^3(x \cdot y) +}\sqrt{|\ln x|}$	1	2	0,5	0,5	1,5	0,5
10	Сумму положительных и произведение отрицательных значений функции $\mathbf{Z = y \cdot tg^2 x - x \cdot e^{-y}}$	1	1,6	0,3	1	2	0,5
11	Произведение отрицательных значений функции $\mathbf{Z = y \cdot tg^2 x - x \cdot e^{-y}}$	0,5	0,7	0,1	0,5	1,5	0,5
12	Сумму положительных и сумму отрицательных значений функции $\mathbf{Z = x \cdot \cos^3 y - y}\sqrt{|x|}$	–1	3	2	0,5	1,5	0,5
13	Сумму отрицательных и количество положительных значений функции $\mathbf{Z =}\sqrt{|\cos y|} \mathbf{- y \cdot \sin^2 x}$	–1	2	1	0,5	1,5	0,25

№	Задача	Исходные данные					
		X_0	X_k	ΔX	Y_0	Y_k	ΔY
14	Произведение положительных и произведение отрицательных значений функции $Z = x \cdot \sin^2 y - y\sqrt{\lvert \cos y \rvert}$	0,3	0,7	0,2	0,5	1,5	0,5
15	Среднее арифметическое значение функции. Определить знак этой величины. $Z = y \cdot \sin^3 x - x\sqrt{\lvert y \rvert}$	0,1	0,5	0,2	−1	3	2
16	Какие значения функции в области преобладают: положительные или отрицательные? $Z = y \cdot tg^2 x - x\sqrt{\lvert y \rvert}$	0,5	1,5	0,5	−3	3	2
17	Произведение положительных и произведение всех значений функции $Z = x \cdot \ln^2 x - ye^{-x}$	1	2	0,5	1	1,2	0,1
18	Сумму значений функции. Какого знака эта величина? $Z = \sin^2(x \cdot y) - \sqrt{\lvert \ln x - y \rvert}$	1	1,4	0,2	1	2	0,5
19	Сумму и произведение значений функции. Какая из этих величин больше? $Z = \cos^2 3x + \sqrt{\lvert \ln x - y \rvert}$	1,2	1,6	0,2	1	2	0,5
20	Отношение произведения отрицательных значений функции к их количеству. $Z = e^{-2y} - \sqrt{\lvert x^3 - \sin^2 y \rvert}$	0	1	0,5	1,1	1,5	0,2

Таблица 3.8 Исследование функции двух переменных при фиксированном значении аргумента

Пример выполнения лабораторной работы – стр.39-42
Математические функции – табл. 5.1 (стр.136)
Порядок выполнения лабораторной работы – раздел 1 (стр.4)

Составить схему и программу решения для следующей задачи:

Вычислить и напечатать таблицу значений функции Z = f(X,Y) и аргументов X,Y в заданной области при условии, что:

- аргумент X изменяется в интервале $X_0 \le X \le X_k$ с шагом ΔX;
- аргумент Y изменяется в интервале $Y_0 \le Y \le Y_k$ с шагом ΔY;

Задания для лабораторной работы 8

При решении используются алгоритмы вычисления суммы, произведения, количества, наибольшего (наименьшего) значения.

Протабулировать функцию и найти:

(п.1 – п.10)

Для каждого фиксированного значения аргумента X: *параметр внешнего цикла – переменная X, а внутреннего – переменная Y.*

№	Задача	Исходные данные					
		X_0	X_k	ΔX	Y_0	Y_k	ΔY
1	Произведение положительных значений функции $Z = x \cdot \sin^2 y - \sqrt{\lvert x^2 - \cos y \rvert}$	1	1,2	0,1	0,5	2	0,5
2	Сумму отрицательных значений функции $Z = x \cdot tg^2 y - y\sqrt{\lvert \ln x \rvert}$	0,2	0,4	0,1	1	2	0,5
3	Наименьшее из положительных значений функции $Z = \sin^2 x - \sqrt{\lvert y \bullet \cos x \rvert}$	1	1,2	0,1	0,5	2	0,5
4	Среднее арифметическое отрицательных значений функции $Z = y \cdot tg^2 x - x\sqrt{\lvert \ln y \rvert}$	0,1	0,3	0,1	0,5	2	0,5
5	Количество положительных значений функции $Z = \sin^2(x \cdot y) - \sqrt{\lvert x \bullet \ln y \rvert}$	1	1,6	0,3	1	2	0,5
6	Наибольшее из отрицательных значений функции $Z = y \cdot \cos^2 x - x\sqrt{\lvert \ln y \rvert}$	1	1,2	0,1	1	2	0,5
7	Сумму и произведение положительных значений функции $Z = tg^2 x - e^{2y}$	0,5	0,7	0,1	–1	1	1
8	Произведение и количество отрицательных значений функции $Z = y \cdot \ln^2 \lvert x \rvert - e^{-y}$	2	3	0,5	–1	1	1
9	Сумму положительных и произведение отрицательных значений функции $Z = tg^2 y - e^{3x}$	0,1	0,3	0,1	0,5	1,5	0,5
10	Сумму и количество положительных значений функции $Z = x \cdot \ln^2 y - e^{-2x}$	0,4	0,6	0,1	2	3	0,5

Протабулировать функцию и найти:

(п.11 – п.20)

Для каждого фиксированного значения аргумента Y: *параметр внешнего цикла – переменная Y , а внутреннего – переменная X.*

№	Задача	Исходные данные					
		X_0	X_k	ΔX	Y_0	Y_k	ΔY
11	Сумму положительных значений функции $Z = \sin^2(x \cdot y) - x\sqrt{\lvert \cos y \rvert}$	1	1,2	0,1	1,9	2,1	0,1
12	Произведение отрицательных значений функции $Z = y \cdot tg^2 x - x\sqrt{\lvert \ln y \rvert}$	0,2	0,6	0,2	1,5	2,1	0,3
13	Наименьшее из положительных значений функции $Z = tg^2 x - e^{2y}$	0,5	1,5	0,5	0,1	0,3	0,1
14	Количество отрицательных значений функции $Z = \ln^2 x - e^{-y}$	1	2	0,5	1	1,2	0,1
15	Среднее арифметическое положительных значений функции $Z = \sin^2 x - \sqrt{\lvert \ln y \rvert}$	1	1,4	0,2	1,9	2,1	0,1
16	Наибольшее из отрицательных значений функции $Z = x \cdot \ln^2 y - ye^{-x}$	0,9	1,1	0,1	2,5	2,7	0,1
17	Произведение и сумму положительных значений функции $Z = y \cdot tg^2 x - e^{x \cdot y}$	1	2	0,5	0,1	0,3	0,1
18	Сумму положительных и сумму отрицательных значений функции $Z = tg^2(x \cdot y) - y \cdot e^x$	0,5	1,5	0,5	1	1,4	0,2
19	Сумму и количество положительных значений функции $Z = y \cdot tg^2 x - x \cdot e^{2y}$	0,5	1,5	0,5	0,1	0,3	0,1
20	Сумму отрицательных и количество положительных значений функции $Z = y \cdot tg^3 x - x \cdot e^y$	–3	–1	1	0,5	0,7	0,1

Таблица 3.9 Вычисление координат точки, наиболее удалённой от начала координат (ближайшей к нему)

Пример выполнения лабораторной работы – стр.43-47
Математические функции – табл. 5.1 (стр.136)
Порядок выполнения лабораторной работы – раздел 1 (стр.4)

Составить схему и программу решения для следующей задачи.

На плоскости (x, y) заданы точки $\{M_1(X_1, Y_1), \ldots, M_n(X_n, Y_n)\}$ и круг радиуса **R** с центром в начале координат, ***n = 10*** – количество точек. Координаты точек задаются в виде массивов абсцисс $(\mathbf{X_1, \ldots, X_n})$ и ординат $(\mathbf{Y_1, \ldots, Y_n})$, *в соответствии с условием задачи* и так, чтобы *искомая точка была одна*.

При решении задачи используются типовые циклические алгоритмы вычисления наибольшего (наименьшего) значения.

Задания для лабораторной работы 9

Определить координаты:

(п.1 – п.10)

Точки, наиболее удалённой от начала координат и лежащей:

№	Задача
1	В первом квадранте и внутри круга
2	Во втором или четвертом квадранте и вне круга
3	В нижней полуплоскости и внутри круга
4	В третьем квадранте и вне круга
5	Во втором или третьем квадранте и внутри круга
6	Во втором квадранте и вне круга
7	В первом или третьем квадранте и внутри круга
8	В верхней полуплоскости и вне круга
9	В четвертом квадранте и внутри круга
10	В третьем или четвёртом квадранте и вне круга

(п.11 – п.20)

Точки, ближайшей к началу координат и лежащей:

11	Во втором квадранте и внутри круга
12	В первом или втором квадранте и вне круга
13	В первом квадранте и внутри круга
14	В левой полуплоскости и вне круга
15	В первом или третьем квадранте и внутри круга
16	Во втором или четвёртом квадранте и вне круга
17	В третьем квадранте и внутри круга
18	В третьем или четвёртом квадранте и вне круга
19	В четвертом квадранте и внутри круга
20	В первом квадранте и вне круга

Таблица 3.10 Вычисление координат искомой точки, ближайшей к заданной (наиболее удалённой от неё)

Пример выполнения лабораторной работы – стр.48-52
Математические функции – табл. 5.1 (стр.136)
Порядок выполнения лабораторной работы – раздел 1 (стр.4)

Составить схему и программу решения для следующей задачи.

На плоскости (x, y) заданы точки$\{M_1(X_1, Y_1), \ldots, M_n(X_n, Y_n)\}$ и $Q(X_q, Y_q)$. Координаты точек задаются в виде массивов абсцисс **$(X_1, \ldots, X_n)$** и ординат **$(Y_1, \ldots, Y_n)$,** *в соответствии с условием задачи* и так, чтобы *искомая точка была одна*.

При решении задачи используется типовой алгоритм вычисления наибольшего (наименьшего) значения.

Задания для лабораторной работы 10

Определить координаты:

(п.1 – п.10)

Точки, ближайшей к точке Q (X_q,Y_q) и лежащей:

№	Задача	n
1	В одной полуплоскости с ней (левой или правой)	8
2	В первом квадранте	9
3	Во втором или четвертом квадранте	7
4	В третьем квадранте	9
5	В четвертом квадранте	8
6	В верхней полуплоскости	9
7	В первом или четвертом квадранте	8
8	Во втором квадранте	7
9	В одном квадранте с точкой Q (X_q,Y_q)	9
10	Во втором или третьем квадранте	8

(п.11 – п.20)

Точки, наиболее удалённой от точки Q (X_q,Y_q) и лежащей:

11	В одной полуплоскости с ней (верхней или нижней)	8
12	В четвертом квадранте	7
13	В правой полуплоскости	9
14	Во втором квадранте	7
15	В первом или втором квадранте	8

16	В третьем квадранте	8
17	Во втором или четвертом квадранте	7
18	В первом квадранте	9
19	В нижней полуплоскости	7
20	Во втором или третьем квадранте	8

Таблица 3.11 Вычисление координат искомой точки

Пример выполнения лабораторной работы – стр.52-55
Математические функции – табл. 5.1 (стр.136)
Порядок выполнения лабораторной работы – раздел 1 (стр.4)

Составить программу решения следующей задачи.

На плоскости (x, y) заданы точки $\{M_1(X_1, Y_1), \ldots, M_n(X_n, Y_n)\}$, *n* – количество точек. Координаты точек задаются в виде массивов абсцисс $\mathbf{(X_1, \ldots, X_n)}$ и ординат $\mathbf{(Y_1, \ldots, Y_n)}$, *в соответствии с условием задачи* и так, чтобы *искомая точка была одна*.

При решении задачи используется типовой алгоритм вычисления наибольшего (наименьшего) значения.

Задания для лабораторной работы 11

Определить координаты:

(п.1 – п.5)

Точки с наименьшей абсциссой среди точек, лежащих:

№	Задача	n
1	В первом и третьем квадранте	8
2	Во втором квадранте	9
3	В третьем и четвертом квадранте	7
4	В третьем квадранте	9
5	В верхней полуплоскости	8

(п.6 – п.10)

Точки с наименьшей ординатой среди точек, лежащих:

6	Во втором и третьем квадранте	9
7	Во втором квадранте	8
8	В правой полуплоскости	7
9	В первом и третьем квадранте	9
10	В четвертом квадранте	8

(п.11 – п.15)

Точки с наибольшей абсциссой среди точек, лежащих:

11	В первом и втором квадранте	8
12	В нижней полуплоскости	7
13	В третьем квадранте	9
14	Во втором и четвертом квадранте	7
15	В четвертом квадранте	8

(п.16 – п.20)

Точки с наибольшей ординатой среди точек, лежащих:

16	В третьем квадранте	8
17	В первом и четвертом квадранте	7
18	В левой полуплоскости	9
19	В первом квадранте	7
20	Во втором и четвертом квадранте	8

Таблица 3.12 Вычисление значения выражения

Пример выполнения лабораторной работы – стр.56-59

Математические функции – табл. 5.1 (стр.136)

Порядок выполнения лабораторной работы – раздел 1 (стр.4)

Составить схему алгоритма и программу решения задачи:

Вычислить значение величины ***Z*** по заданной формуле. Массив $\mathbf{A_i, (i = 1,\ldots,n)}$ задаётся самостоятельно. При вычислении min, max параметр цикла ***i*** изменяется в интервале **[1;*n*]**.

При решении используются типовые алгоритмы вычисления суммы, произведения, наибольшего (наименьшего) значения.

Задания для лабораторной работы 12

№	Формула для вычисления значения Z	n
1	$Z = \sum_{i=2}^{n-1} (i! + A_i) - \min_i \left(A_i \prod_{i=1}^{n-2} \sqrt{i} \right)$	6
2	$Z = \min_i \left(A_i \prod_{i=1}^{n-3} i^2 \right) + \sum_{i=3}^{n} A_i^2$	6

3	$Z=\prod_{i=1}^{n-2}\left(A_i+\sum_{i=2}^{n-1}i!\right)-\min_i(iA_i)$	6
4	$Z=\max_i(iA_i)+\sum_{i=2}^{n-2}A_i^i-\prod_{i=1}^{n-3}i!$	6
5	$Z=\sum_{i=2}^{n-1}\left(i^2-\prod_{i=1}^{n-3}A_i\right)+\max_i\left(A_i^2\right)$	6
6	$Z=\max_i\left(A_i^2-\prod_{i=3}^{n-1}A_i\right)+\sum_{i=1}^{n-2}\sqrt{i}$	7
7	$Z=\sum_{i=2}^{n-3}\sqrt{i!}+\min_i\lvert A_i\rvert-\prod_{i=1}^{n-3}A_i$	7
8	$Z=\min_i\left(A_i\sqrt{i}\right)+\sum_{i=2}^{n-2}\left(i!-\prod_{i=1}^{n-3}A_i^2\right)$	7
9	$Z=\prod_{i=2}^{n-3}\left(i!-\min_i\{iA_i\}\right)+\sum_{i=1}^{n-2}\sqrt{i}A_i$	7
10	$Z=\max_i\left(A_i\sum_{i=2}^{n-3}i!\right)-\prod_{i=1}^{n-2}A_i^{-i}$	7
11	$Z=\sum_{i=2}^{n-3}\left(\ln i-\prod_{i=1}^{n-2}A_i\right)+\min_i\left(\sqrt{\lvert A_i\rvert}\right)$	6
12	$Z=\max_i\left(iA_i^2\right)-\sum_{i=1}^{n-2}\left(i!+\prod_{i=2}^{n-3}A_i\right)$	6
13	$Z=\prod_{i=1}^{n-2}\left(\sqrt{i}-\min_i\left\{A_i^3\right\}\right)+\sum_{i=2}^{n-1}A_i$	6
14	$Z=\min_i(i!A_i)+\prod_{i=2}^{n-3}\left(A_i-\sum_{i=1}^{n-2}\sqrt{i}\right)$	6
15	$Z=\sum_{i=1}^{n-2}\left(i!+\max_i\lvert A_i\rvert\right)-\prod_{i=2}^{n-1}\sqrt{\lvert A_i\rvert}$	6

16	$Z = \max_i \left(iA_i^2\right) \sum_{i=1}^{n-1} \left(\sqrt{i!} - \prod_{i=2}^{n-2} A_i \right)$	5
17	$Z = \prod_{i=2}^{n-1} \left(\sqrt{i!} - \max_i \lvert A_i \rvert \right) + \sum_{i=1}^{n} A_i^2$	5
18	$Z = \min_i \left(A_i + \prod_{i=2}^{n-1} i! \right) + \sum_{i=1}^{n-2} A_i^2$	5
19	$Z = \sum_{i=2}^{n} \left(\ln i + \max_i \{ i! A_i \} \right) - \prod_{i=2}^{n} A_i$	5
20	$Z = \max_i \left(\lvert A_i \rvert - \prod_{i=2}^{n-1} A_i^2 \right) + \sum_{i=1}^{n-2} iA_i$	5

Таблица 3.13 Анализ эффективности работы предприятия

Пример выполнения лабораторной работы – стр.59-63
Порядок выполнения лабораторной работы – раздел 1 (стр.4)

Составить схему алгоритма и программу решения задачи:

Выполнение плана (в процентах) по объемам производства на судостроительном предприятии в течение года и названия месяцев представлены массивами $\mathbf{A_i}$ и $\mathbf{MC_i}$ **(i = 1,...,12)**. План – 100%, менее 100% – план не выполнен, более 100% – перевыполнен.

Исходные данные

1-е полугодие						
Месяц	1	2	3	4	5	6
	Январь	Февраль	Март	Апрель	Май	Июнь
%	$\mathbf{A_1}$	$\mathbf{A_2}$	$\mathbf{A_3}$	$\mathbf{A_4}$	$\mathbf{A_5}$	$\mathbf{A_6}$
	101	**98**	**100**	**108**	**97**	**105**
2-е полугодие						
Месяц	7	8	9	10	11	12
	Июль	Август	Сентябрь	Октябрь	Ноябрь	Декабрь
%	$\mathbf{A_7}$	$\mathbf{A_8}$	$\mathbf{A_9}$	$\mathbf{A_{10}}$	$\mathbf{A_{11}}$	$\mathbf{A_{12}}$
	99	**107**	**96**	**112**	**100**	**98**

При решении задачи используются типовые алгоритмы вычисления суммы, количества, наименьшего (наибольшего) значения.

Вычисление среднегодового процента выполнения плана предусмотреть в программе.

Задания для лабораторной работы 13

Определить:

№	Задача
1	Средний процент выполнения плана в течение тех месяцев, когда план не выполнялся
2	Полугодие с наибольшим средним процентом выполнения плана
3	Месяцы с процентом выполнения, превышающим среднегодовой
4	В каком полугодии достигнут наибольший процент?
5	Средний процент выполнения плана в течение тех месяцев, когда план перевыполнялся
6	Месяц с наибольшим процентом перевыполнения плана
7	Превышает ли среднегодовой процент выполнения плана 100%? Вывести соответствующее текстовое сообщение.
8	Количество месяцев с процентом выполнения плана ниже среднегодового
9	Наибольший процент выполнения плана в течение тех месяцев, когда план не выполнялся
10	Количество месяцев, в течение которых план перевыполнялся
11	Наименьший процент выполнения плана в течение тех месяцев, когда план перевыполнялся
12	В течение каких месяцев план не выполнялся
13	Месяц с наибольшим отставанием от плана в течение года
14	Наибольший процент выполнения плана в каждом полугодии. В каком полугодии этот показатель выше?
15	Для каждого полугодия количество месяцев, когда план не выполнялся. В каком полугодии этот показатель ниже?
16	Месяц с наибольшим процентом среди месяцев, в течение которых процент выполнения плана не превышает среднегодовой.
17	В каком полугодии наблюдается самый низкий процент выполнения?
18	В каком полугодии средний процент выполнения плана выше среднегодового?
19	Месяцы с процентом выполнения плана ниже среднего
20	Каких месяцев в году больше: с перевыполнением или невыполнением плана?

Таблица 3.14 Анализ итогов экзамена в группе

Пример выполнения лабораторной работы – стр.64-67
Порядок выполнения лабораторной работы – раздел 1 (стр.4)

Составить схему алгоритма и программу решения задачи:
Результаты сдачи экзамена и фамилии студентов группы представлены массивами $\mathbf{A_i}$ и $\mathbf{F_i}$ **(i = 1,…,10)**.

Исходные данные

Студент	F_1	F_2	F_3	F_4	F_5
Фамилия	Юдин И.В.	Ашин С.В.	Итин К.Л.	Бут С.С.	Бех Л.Л.
Оценка	A_1	A_2	A_3	A_4	A_5
	5	4	4	3	2
Студент	F_6	F_7	F_8	F_9	F_{10}
Фамилия	Этин П.Н.	Юров Ю.Ю.	Апин Р.П.	Фет В.И.	Кац Д.Е.
Оценка	A_6	A_7	A_8	A_9	A_{10}
	5	4	2	4	3

При решении задачи используются типовой алгоритм вычисления суммы и количества.

Задания для лабораторной работы 14

Вычисление среднего балла предусмотреть в программе!

Определить:

№	Задача
1	Какая категория (хороших или отличных) оценок преобладает?
2	Процент хороших и отличных оценок в группе
3	Количество успевающих студентов
4	Процент оценок каждой категории
5	Количество и фамилии студентов с оценкой «удовлетворительно»
6	Процент и фамилии успевающих студентов
7	Средний балл среди хорошистов и отличников
8	Процент неудовлетворительных оценок
9	Каких оценок в группе больше: «хорошо» и «отлично» или «удовлетворительно»?
10	Средний балл среди успевающих студентов
11	Фамилии и количество неуспевающих студентов
12	Процент успевающих студентов и процент отличников

№	Задача
13	Процент и фамилии студентов-хорошистов
14	Количество успевающих и количество неуспевающих студентов. Какая категория преобладает?
15	Превышает ли средний балл группы оценку «хорошо»? Вывести текстовое сообщение.
16	Фамилии успевающих студентов с оценкой ниже среднего балла
17	Процент оценок, превышающих средний балл группы
18	Количество студентов из числа успевающих с оценкой ниже среднего балла группы
19	Фамилии студентов с оценкой выше среднего балла группы
20	Каких студентов из числа успевающих больше: получивших оценку выше или ниже среднего балла группы?

Таблица 3.15 Анализ эффективности работы флота

Пример выполнения лабораторной работы – стр.70-80

Составить схему алгоритма и программу решения задачи:

Фактические данные по объёмам перевозок грузов пятью судами за четыре месяца навигации представлены в виде матрицы A(5x4), где строка – судно, столбец - месяц. Данные приведены в процентном отношении к нормативным объёмам перевозок (плану). За план принимается значение, равное 100%. Менее 100% – план не выполнен, более 100% – перевыполнен.

Матрица задаётся в соответствии с условием задачи. Элемент матрицы A_{ij} – это процент выполнения плана судном с номером i за месяц навигации с номером j.

Порядок выполнения лабораторной работы:

1. В соответствии с обходом матрицы [по строкам/столбцам] определить параметры внешнего и внутреннего цикла (**i** или **j**).
2. Определиться с типом алгоритма решения задачи – вычисления суммы, количества, наибольшего или наименьшего значения.
3. В соответствии с алгоритмом при необходимости дополнительно к исходным данным ввести обозначения переменных.
4. Представить имена всех переменных в таблице.
5. Выполнить ручной расчёт результатов.
6. Составить макет печати исходных данных и результатов.

7. Составить схему алгоритма решения задачи.

8. В соответствии со схемой и макетом печати написать программу.

9. Проанализировать полученные результаты. Ручной и машинный расчёты должны совпадать.

Примечание. Ввод матрицы можно осуществлять как по строкам, так и по столбцам. Вывод же матрицы организуется в общепринятом виде, т.е. по строкам. Матрица, распечатанная по столбцам, является транспонированной, и не будет соответствовать исходной.

При решении задач используются типовые алгоритмы вычисления суммы, количества, наибольшего (наименьшего) значения.

Задания для лабораторной работы 15

***Определить*:**

(п.1 – п.10)

№	Задача
1	Количество судов, ежемесячно выполнявших план
2	Месяц навигации с наименьшим средним процентом
3	Суда, ежемесячно перевыполнявшие план (100%)
4	Количество месяцев, в которых план выполнялся всеми судами
5	Превышает ли средний процент выполнения плана судами флота за навигацию 100%?
6	Судно с наименьшим средним процентом выполнения плана
7	Месяцы навигации, в которых план не выполнялся ни одним судном
8	Наименьший процент за навигацию. Превышает ли он 100%?
9	Судно с наименьшим средним процентом среди судов, ежемесячно выполнявших план
10	Судно с наибольшим процентом перевыполнения плана

(п.11 – п.15)

Для каждого судна (строки i):

№	Задача
11	Количество месяцев, в течение которых план перевыполнялся
12	Превышает ли средний процент выполнения плана 100%?
13	Наибольший процент в течение месяцев, когда план не выполнялся
14	Средний процент в течение месяцев, когда план перевыполнялся
15	Наименьший процент в течение месяцев, когда план выполнялся

(п.16 – п.20)

Для каждого месяца (столбца j):

№	Задача
16	Наименьший процент среди судов, перевыполнявших план
17	Превышает ли средний процент выполнения плана 100%?
18	Количество судов, не выполнивших план
19	Средний процент среди судов, перевыполнивших план
20	Наибольший процент среди судов, не выполнивших план

Таблица 3.16 Анализ итогов сессии в группе

Пример выполнения лабораторной работы – стр.81-85
Порядок выполнения лабораторной работы – стр.4, стр.120

Составить схему алгоритма и программу решения задачи:

Результаты сдачи сессии из четырёх предметов в группе из пяти студентов представлены в виде матрицы A(5x4), где строка – студент, столбец – предмет. Элемент матрицы A_{ij} – оценка, полученная студентом с номером i по предмету с номером j. Матрица задаётся в соответствии с условием задачи.

При решении задач используются типовые алгоритмы вычисления суммы, количества, наибольшего (наименьшего) значения.

Задания для лабораторной работы 16

Определить для группы:

(п.1 – п.10)

№	Задача
1	Процент успевающих студентов
2	Превышает ли средний балл группы оценку «хорошо»?
3	Количество отличников в группе
4	Процент хороших и отличных оценок
5	Какая из оценок [«хорошо» или «удовлетворительно»] преобладает?
6	Количество студентов, сдавших на стипендию (критерий – без троек)
7	Предмет с наименьшим средним баллом
8	Количество студентов, получивших оценку «неудовлетворительно»
9	Процент неудовлетворительных оценок в группе
10	Количество студентов, сдавших все экзамены на оценку «удовлетворительно»

(п.11 – п.15)

Определить для каждого предмета (столбца j):

№	Задача
11	Средний балл среди успевающих студентов
12	Процент неудовлетворительных оценок. Предусмотреть вывод текстового сообщения, если таких оценок нет
13	Превышает ли средний балл оценку «удовлетворительно»?
14	Количество хороших и отличных оценок
15	Какая категория оценок преобладает: «отлично» и «хорошо» или «удовлетворительно»?

(п.16 – п.20)

Определить для каждого студента (строки i):

№	Задача
16	Количество удовлетворительных оценок. Если таких оценок нет, вывести текстовое сообщение
17	Превышает ли средний балл оценку «хорошо»?
18	Каких оценок получено больше: «отлично» и «хорошо» или «удовлетворительно»?
19	Процент хороших и отличных оценок
20	Процент неудовлетворительных оценок. Предусмотреть вывод текстового сообщения, если таких оценок нет

Таблица 3.17 Формирование одномерного массива

Пример выполнения лабораторной работы – стр.85-90

Порядок выполнения лабораторной работы –стр.4, стр.120

Составить схему алгоритма и программу решения задачи.

При решении используются алгоритмы вычисления суммы, произведения, количества, наибольшего (наименьшего) значения.

Задания для лабораторной работы 17

(п.1 – п.10)

Дана квадратная матрица **A(6x6)**.

Сформировать массив **(B_1, …, B_6)** в соответствии с условием задачи. Упорядочить массив по возрастанию, либо по убыванию (схема алгоритма – рис. 2.31, стр.89). Матрица задаётся самостоятельно.

Каждый элемент массива определяется как разность между:

№	Задача
1	Наибольшим и наименьшим элементами строки матрицы. Найти среднее арифметическое элементов массива B
2	Наименьшим и диагональным элементами строки матрицы. Найти произведение элементов массива B
3	Диагональным и наибольшим элементами строки матрицы. Найти сумму отрицательных элементов массива B
4	Количеством положительных и количеством отрицательных элементов строки матрицы. Найти наибольший элемент массива B
5	Суммой положительных и произведением отрицательных элементов строки матрицы. Найти наименьший элемент массива B
6	Диагональным и наименьшим элементами столбца матрицы. Найти сумму отрицательных элементов массива B
7	Наибольшим и диагональным элементами столбца матрицы. Найти произведение положительных элементов массива B
8	Количеством положительных и суммой отрицательных элементов столбца матрицы. Найти наименьший элемент массива B
9	Наибольшим из отрицательных и диагональным элементами столбца матрицы. Найти сумму элементов массива B
10	Суммой отрицательных и произведением положительных элементов столбца матрицы. Найти наибольший элемент массива B

(п.11 – п.20)

По матрице сформировать массив (B_1, …, B_6). Причём, формула для вычисления элемента массива зависит от чётности строки, либо столбца матрицы. Чётность проверяется условием:

$$\mathbf{(-1)^i > 0} \text{ или } \mathbf{(-1)^j > 0}$$

Если условие выполняется (да), строка/столбец – чётные, иначе – нечётные.

Каждый элемент массива определяется как:

№	Задача
11	Диагональный элемент в чётной и наименьший элемент в нечётной строке матрицы. Найти среднее арифметическое отрицательных элементов массива B
12	Наибольший элемент в чётной и диагональный элемент в нечётной строке. Найти произведение положительных элементов массива B

13	Количество положительных элементов в чётной и сумма отрицательных элементов в нечётной строке матрицы. Найти наименьший элемент массива B
14	Первый элемент в чётной и наибольший из отрицательных элементов в нечётной строке матрицы. Найти сумму элементов массива B
15	Наименьший элемент в чётной и наибольший – в нечётной строке матрицы. Найти количество положительных элементов массива B
16	Диагональный элемент в чётном и количество положительных элементов в нечётном столбце. Найти наименьший элемент массива B
17	Сумма отрицательных элементов в чётном и диагональный элемент в нечётном столбце матрицы. Найти наибольший элемент массива B
18	Наименьший из положительных элементов в чётном и наибольший из отрицательных элементов в нечётном столбце матрицы. Найти сумму элементов массива B
19	Первый элемент в чётном и наибольший элемент в нечётном столбце матрицы. Найти количество положительных элементов массива B
20	Произведение положительных элементов в чётном и наименьший элемент в нечётном столбце. Найти сумму элементов массива B

4. Задания для самостоятельной работы

Лабораторная работа 4.1

Вычисление значения функции

Тема. Линейный алгоритм

Математические функции – табл. 5.1 (стр.136)

Пример выполнения лабораторной работы – стр.5-8

Составить программу вычисления функции Y(X). Значения X и Y(X) приведены в таблице. При несовпадении результата необходимо исправить ошибки и повторно выполнить программу.

Задания для лабораторной работы 4.1

№	Формула для вычисления функции	Аргумент X	Функция Y
1	$y = \cos^2(3 \cdot x) + \sqrt{\ln\lvert x\rvert} \cdot e^x$	1,4	2.5926
2	$y = \sqrt[3]{\lvert x\rvert} + \sin^2(x) \cdot e^{-2x}$	1,5	1.1943
3	$y = \ln^2\lvert x \cdot \cos 8^0\rvert + \sqrt{\lvert x\rvert} \cdot e^x$	0,8	2.0448
4	$y = \sqrt[3]{\lvert x \cdot \cos 2^0\rvert} + tg^2(x \cdot \ln x)$	0,6	0.9434

№	Формула для вычисления функции	Аргумент X	Функция Y
5	$y = \ln^3(x \cdot \sqrt{\lvert \sin 5^0 \rvert}) - e^{-x}$	2,5	–0.1101
6	$y = \cos^2(x) \cdot e^{-x} + \sqrt{\ln\lvert x \rvert}$	2,7	1.0516
7	$y = arctg^2 x - \sqrt{\lvert x \rvert} \cdot e^{-x}$	0,5	–0.2139
8	$y = \sqrt{\lvert x \cdot \sin 3^0 \rvert} + tg^3(x \cdot e^x)$	0,3	0.2041
9	$y = \sin^2(3 \cdot x) - \ln(x \cdot \sqrt{\lvert x \rvert})$	1,4	0.2549
10	$y = \ln^2(e^{-x}) + \sqrt[3]{\lvert x \cdot \cos 7^0 \rvert}$	0,5	1.0417
11	$y = \sqrt{\ln\lvert x \rvert} \cdot e^{2x} - arctg^2 x$	1,2	3.9393
12	$y = \sin^2(x \cdot e^{-x}) + \sqrt{\lvert x \cdot tg 5^0 \rvert}$	0,7	0.3635
13	$y = x \cdot e^{3x} - arctg^2(\sqrt{\lvert x \rvert})$	–2	–0.9175
14	$y = \cos^2(x \cdot \ln\lvert x \rvert) + e^{-2x}$	2,3	0.1243
15	$y = \sqrt[3]{\lvert x \cdot \ln x \rvert} + ch^2 x$	0,5	1.9739

Лабораторная работа 4.2

Вычисление функции, заданной в виде ветвления

Тема. Разветвлённый алгоритм

Математические функции – табл. 5.1 (стр.136)

Примеры выполнения лабораторных работ – стр.8-14

Составить схему и программу вычисления функции y(x).

Для выполнения лабораторной работы необходимо:

1. В таблице даны значения аргумента x и функции y(x) для каждой ветви. При выполнении программы по запросу с экрана ввести с клавиатуры первое значение аргумента x. Полученное значение y(x) сравнить с приведённым в таблице.

2. Действие п.1 выполняется поочерёдно для каждой ветви.

3. При несовпадении результата повторить действие п.1.

Задания для лабораторной работы 4.2

№	Функции y(x)	Значение	
		x	y(x)
1	$y=\begin{cases}3\sqrt{\lvert x-1\rvert}-3.25 & \text{при} \quad x<-1.25\\ -x & \text{при} \quad -1.25\le x\le 1.25\\ 3.25-3\sqrt{\lvert x-1\rvert} & \text{при} \quad x>1.25\end{cases}$	–2 1 2	1,95 –1 0,25
2	$y=\begin{cases}\sin^2(x+1) & \text{при} \quad x<-1\\ -(1+\cos\pi x) & \text{при} \quad -1\le x<1\\ \sin^2(x-1) & \text{при} \quad x\ge 1\end{cases}$	–2 0 3	0,7 –2 0,8
3	$y=\begin{cases}-\cos 5\pi(x-0.2) & \text{при} \quad x<0.2\\ 1-2\cos 5\pi(x-.2) & \text{при} \quad 0.2\le x<0.4\\ 3 & \text{при} \quad x\ge 0.4\end{cases}$	0 0,3 1	1 1 3
4	$y=\begin{cases}1+2x^2 & \text{при} \quad x\le 0\\ 1 & \text{при} \quad 0<x<0.5\\ 1+(x-0.5)^4 & \text{при} \quad x\ge 0.5\end{cases}$	–1 0,2 0,5	3 1 1
5	$y=\begin{cases}6\sqrt[3]{\lvert x\rvert}-5 & \text{при} \quad x<-1\\ x^2 & \text{при} \quad -1\le x\le 1\\ 6\sqrt[3]{x}-5 & \text{при} \quad x>1\end{cases}$	–8 –1 8	7 1 7
6	$y=\begin{cases}7\sqrt{\lvert x-1\rvert}-11 & \text{при} \quad x<-3\\ x & \text{при} \quad -3\le x<3\\ 7\sqrt{\lvert x+1\rvert}-11 & \text{при} \quad x\ge 3\end{cases}$	–8 1 3	10 1 3
7	$y=\begin{cases}\pi-(x+\pi)^2 & \text{при} \quad x<-\pi\\ \pi+\cos x+1 & \text{при} \quad -\pi\le x<\pi\\ \pi+(x-\pi)^2 & \text{при} \quad x\ge\pi\end{cases}$	–2π 0 2π	–6,73 5,14 13,01
8	$y=\begin{cases}0 & \text{при} \quad x<1\\ \ln x/x & \text{при} \quad 1\le x<e\\ x^2/e^x & \text{при} \quad x\ge e\end{cases}$	–1 2 3	0 0,34 0,45
9	$y=\begin{cases}1+\sin^3(x+0.5) & \text{при} \quad x<-0.5\\ 1 & \text{при} \quad -0.5\le x\le 0.5\\ 1+\sin^3(x-0.5) & \text{при} \quad x>0.5\end{cases}$	–1 0 1	0,89 1 1,1

№	Функции y(x)	Значение	
		x	y(x)
10	$y=\begin{cases} x+e^{-x} & \text{при } x\le 0 \\ 1 & \text{при } 0<x<1 \\ 1+(x-1)^2 & \text{при } x\ge 1 \end{cases}$	0 0,5 2	1 1 2
11	$y=\begin{cases} -\cos(x+1) & \text{при } x\le -1 \\ \sin\frac{\pi}{2}x & \text{при } -1<x<1 \\ \cos(x-1) & \text{при } x\ge 1 \end{cases}$	−1 0,5 1	−1 0,7 1
12	$y=\begin{cases} (x+1)^2 & \text{при } x<-1 \\ 1+\cos\pi x & \text{при } -1\le x<1 \\ -(x-1)^2 & \text{при } x\ge 1 \end{cases}$	−2 0 2	1 2 −1
13	$y=\begin{cases} \lvert x\rvert & \text{при } x\le 0 \\ -\sin 2\pi x & \text{при } 0<x\le 1 \\ 1-x & \text{при } x>1 \end{cases}$	−2 0 2	2 0 −1
14	$y=\begin{cases} x(x+2) & \text{при } x\le -2 \\ \sin\pi x & \text{при } -2<x\le 0 \\ x(x+2) & \text{при } x>0 \end{cases}$	−3 −1 1	3 0 3
15	$y=\begin{cases} \sqrt{\lvert x\rvert} & \text{при } x<0 \\ \sin^2 x+\sqrt{x} & \text{при } 0\le x\le \pi \\ x^2 & \text{при } x>\pi \end{cases}$	−4 1 4	2 1,7 16

Лабораторная работа 4.3

Расчёт грузовой шкалы судна

Тема. Табулирование функции одной переменной

Примеры выполнения лабораторных работ – стр.17-33
Порядок выполнения лабораторной работы – раздел 1 (стр.4)

1. Задание

Составить схему и программу решения следующей задачи:
При ведении погрузочных работ масса груза на судне G определяется по таблице грузового размера (грузовой шкале). Рассчитать зависимость между осадкой судна T и количеством груза G, задаваемую в виде ветвления:

$$G = \begin{cases} \gamma LB\sigma(T - To), \text{если } To \le T \le Tcp; \\ \gamma LB(\sigma T - \sigma oTo), \text{если } Tcp < T \le Tmax. \end{cases}$$

Плотность пресной воды γ = 1, её в расчётах не учитываем.

Исходные данные

L	B	σ	σ_0	T_0	T_{cp}	T_{max}	ΔT	γ
8,5	*13,8*	*0,85*	*0,65*	*0,2*	*0,6*	*1,2*	*0,2*	*1*

2. Решение

Математическая формулировка задачи.

Вычислить и напечатать таблицу значений функции G и аргумента T при условии, что аргумент T изменяется на интервале $T_0 \le T \le T_{max}$ с шагом ΔT. Функция G определяется как:

$$G = \begin{cases} f1(T), \text{если } To \le T \le Tcp; \\ f2(T), \text{если } Tcp < T \le Tmax; \end{cases}$$

где $f1(T) = LB\sigma(T - To)$, $f2(T) = LB(\sigma T - \sigma oTo)$

Обозначения переменных				
Имя переменной		Тип переменной		Пояснение
в задаче	в программе	**Fortran**	**C**	
L, B	**L, B**	**REAL**	**FLOAT**	Длина и ширина судна
σ, σ_0	**S, S0**	**REAL**	**FLOAT**	Коэффициент полноты водоизмещения при текущей осадке и порожнем
T, G	**T, G**	**REAL**	**FLOAT**	Текущее значение аргумента и функции
T_0, T_{max}	**T0, TMAX**	**REAL**	**FLOAT**	Начальное и конечное значение аргумента T (осадка порожнем и полная осадка судна)
T_{cp}	**TCR**	**REAL**	**FLOAT**	Среднее значение аргумента на интервале $To < Tcp < Tmax$
ΔT	**DT**	**REAL**	**FLOAT**	Шаг изменения аргумента T

Алгоритм циклической структуры. Параметр цикла – переменная T (осадка судна). Вывод результатов осуществляется в виде таблицы текущих значений аргумента T (осадки) и функции G (груза).

3. Макет печати результатов

```
T = 0.2          G =   0.0
T = 0.4          G = 199.41
T = 0.6          G = 398.8201
T = 0.8          G = 598.23
T = 1.0          G = 797.6401
T = 1.2          G = 997.05
```

Задания для лабораторной работы 4.3

Груз перевозится по реке. Коэффициент полноты водоизмещения изменяется при осадке больше $T_{ср}$ ($T > T_{ср}$). Рассчитать грузовую шкалу для следующих судов.

Исходные данные								
№	**L**	**B**	$\mathbf{T_0}$	$\mathbf{T_{max}}$	$\mathbf{T_{ср}}$	$\mathbf{\Delta T}$	$\mathbf{\sigma_0}$	$\mathbf{\sigma}$
1	85	13,8	0,5	2,5	1,5	0,2	0,66	0,85
2	90	13	0,8	2,4	1,6	0,2	0,68	0,74
3	85	12,6	0,7	2,5	1,5	0,2	0,66	0,74
4	88	12,6	0,5	2,5	1,5	0,25	0,68	0,78
5	90	12	1	5	2,6	0,4	0,68	0,8
6	84	12,2	0,8	2,8	1,8	0,25	0,69	0,86
7	120	16	0,6	2,8	1,6	0,2	0,62	0,82
8	130	16,5	0,9	2,5	1,5	0,2	0,7	0,84
9	110	13,3	1,1	3,5	2	0,3	0,68	0,78
10	114	13	1,2	6	3,2	0,4	0,68	0,8
11	112	13	0,9	4,2	2,4	0,3	0,66	0,82
12	90	12,6	0,8	2,6	1,8	0,2	0,68	0,81
13	92	14	0,7	2,5	1,5	0,2	0,66	0,82
14	102	12,4	0,6	2,6	1,4	0,2	0,68	0,88
15	96	13	1,2	3,4	2,2	0,2	0,7	0,82

Лабораторная работа 4.4
Вычисление координат точки
Тема. Одномерные массивы: min, max

Математические функции – табл. 5.1 (стр.136)
Примеры выполнения лабораторных работ – стр.43-55

Составить схему и программу решения для следующей задачи:
На плоскости (x, y) заданы точки $\{M_1(X_1, Y_1), \ldots, M_n(X_n, Y_n)\}$, точка $Q(X_q,Y_q)$,а также круг радиуса R с центром в начале координат. Определить координаты точки $M(X_k, Y_k)$ в соответствии с вариантом, k – порядковый номер точки, количество точек n = 10.

Для выполнения задания необходимо:

- задать координаты точек $\{M_1(X_1, Y_1), \ldots, M_n(X_n, Y_n)\}$ и $Q(X_q, Y_q)$, а также радиус круга R *в соответствии с условием задачи*.
- представить точки и круг на плоскости (X;Y).
- найти и обозначить на графике искомую точку $M(X_k, Y_k)$.

Задания для лабораторной работы 4.4

При решении используется алгоритм вычисления наибольшего (наименьшего) значения. **Определить координаты точки:**

(п.1 – п.3)

С наименьшей (наибольшей) абсциссой (ординатой) и лежащей:

№	Задача
1	Вне круга и в одном квадранте с точкой Q
2	В одной полуплоскости (левой или правой) с точкой Q и внутри круга
3	Вне круга и в одной полуплоскости (верхней или нижней) с точкой Q

(п.4 – п.6)

Наиболее удалённой от начала координат (ближайшей к нему) и лежащей:

4	Внутри круга и в одном квадранте с точкой Q
5	В одной полуплоскости (верхней или нижней) с точкой Q и вне круга
6	Внутри круга и в одной полуплоскости (левой или правой) с точкой Q

(п.7 – п.11)

Ближайшей к точке Q (X_q,Y_q) и лежащей:

7	Вне круга и в одном квадранте с точкой Q
8	В первом квадранте и внутри круга
9	В верхней полуплоскости и вне круга
10	Во втором или четвёртом квадранте и внутри круга
11	В третьем квадранте и вне круга

(п.12 – п.15)

Наиболее удалённой от точки Q (X_q,Y_q) и лежащей:

12	Во втором квадранте и вне круга
13	Внутри круга и в одном квадранте с точкой Q
14	В четвёртом квадранте и вне круга
15	В правой полуплоскости и внутри круга

Лабораторная работа 4.5

Вычисление площади живого сечения одноруканого русла

Тема. Одномерные массивы: вычисление суммы

Порядок выполнения лабораторной работы – раздел 1 (стр.4)

1. Задание

Составить схему и программу решения следующей задачи:

Для проектирования выправительных сооружений и дноуглубительных работ при выполнении расчётов возникает необходимость вычисления площади живого сечения русла по экспериментальным данным – результатам измерений. Вычисление площади сечения осуществляется методом трапеции по формуле:

$$S = \sum_{i=1}^{m-1} \frac{(h_i + h_{i+1})}{2} \cdot (x_{i+1} - x_i)$$, где **m** – число замеров (точек);

Результаты измерений представлены в виде массивов x_i и h_i (i = 1,…,m), m = 8.

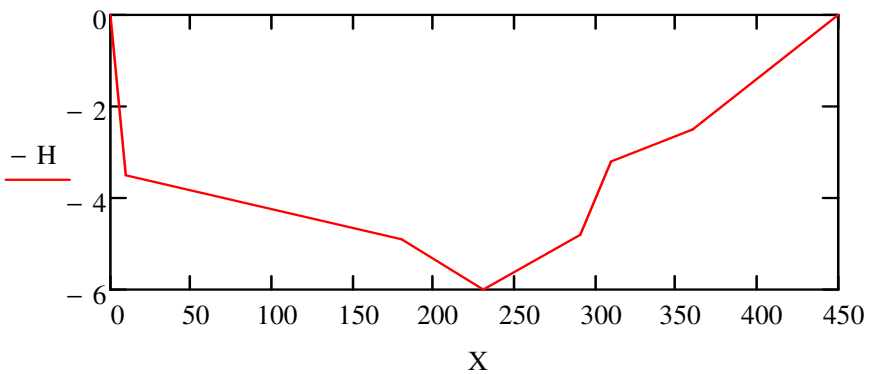
x_i – расстояние от берега до i-ой точки измерения;

h_i – глубина в i-ой точке.

Исходные данные

Координаты точек							
Расстояние от берега до i-ой точки измерения							
X_1	X_2	X_3	X_4	X_5	X_6	X_7	X_8
0	10	180	230	290	310	360	450
Глубина в i-ой точке измерения							
H_1	H_2	H_3	H_4	H_5	H_6	H_7	H_8
0	3,5	4,9	6	4,8	3,2	2,5	0

График сечения дна русла



2. Решение

Обозначения переменных			
Имя переменной	Тип переменной		Пояснение
	Fortran	**C**	
M, i	**INTEGER**	**INT**	Количество точек измерений и номер точки
X_i	**REAL**	**FLOAT**	Расстояние от берега до i-ой точки
H_i	**REAL**	**FLOAT**	Глубина в i-ой точке измерения
S	**REAL**	**FLOAT**	Сумма, площадь сечения русла

При решении используется алгоритм вычисления суммы. Параметр цикла i (номер точки) изменяется в интервале от 1 до 7 с шагом 1. В цикле осуществляется ввод-вывод данных, а также вычисление суммы S.

3. Макет печати

```
                ИСХОДНЫЕ ДАННЫЕ
          ЧИСЛО ТОЧЕК ИЗМЕРЕНИЙ M = 8

          КООРДИНАТЫ ТОЧЕК ИЗМЕРЕНИЙ

     0    10   180   230   290   310   360   450
     0   3.5   4.9     6   4.8   3.2   2.5     0

          ПЛОЩАДЬ СЕЧЕНИЯ РУСЛА = 1663
```

Задания для лабораторной работы 4.5

По результатам замеров вычислить площадь сечения русла. Число точек измерения $m = 8$.

№	Координаты X_i	Глубины H_i
1	0 100 130 180 240 250 310 390	0,0 1,8 1,1 3,5 2,8 2,0 3,5 0,0
2	0 10 50 100 150 210 270 330	0,0 3,2 1,3 2,9 2,8 2,7 4,0 0,0
3	0 90 130 150 170 180 230 240	0,0 2,9 3,6 2,0 2,8 2,7 3,1 0,0
4	0 10 80 140 160 200 240 260	0,0 1,5 3,6 2,5 2,3 1,4 1,4 0,0
5	0 30 120 220 260 290 330 380	0,0 1,7 3,2 3,9 2,8 3,1 2,3 0,0
6	0 10 90 150 180 210 250 350	0,0 1,3 3,0 2,3 1,4 1,6 2,5 0,0
7	0 20 80 160 230 240 260 350	0,0 1,2 1,8 3,1 3,7 1,4 2,2 0,0
8	0 40 90 130 200 250 260 310	0,0 2,3 3,6 1,8 1,6 1,8 2,9 0,0
9	0 90 180 260 310 360 460 560	0,0 1,3 2,7 1,6 1,4 3,7 1,9 0,0
10	0 90 110 180 230 310 390 400	0,0 1,7 2,7 1,4 2,0 3,6 3,9 0,0
11	0 10 80 80 130 190 190 250	0,0 2,2 2,0 3,0 2,8 1,6 3,6 0,0
12	0 50 110 190 220 250 320 410	0,0 2,8 2,6 3,2 1,5 2,6 2,0 0,0
13	0 80 140 200 260 280 360 460	0,0 2,1 1,9 2,6 1,9 3,6 4,0 0,0
14	0 40 130 150 170 270 320 320	0,0 1,7 2,9 1,7 1,6 3,2 1,7 0,0
15	0 80 170 180 270 350 370 390	0,0 1,9 2,2 3,0 2,2 1,9 3,5 0,0

Лабораторная работа 4.6

Вычисление границ искомого интервала

Тема. Двумерный массив (матрица)

Порядок выполнения лабораторной работы – стр.4,120
Примеры выполнения лабораторных работ – стр.70-90

Для выполнения лабораторной работы необходимо:

- задать квадратную матрицу A(n x n), n = 8. Численные значения элементов A_{ij}, где i,j – номер строки и столбца, следует задавать не случайным образом, а подбирать с учётом смысла и условия задачи, что очень важно;
- выполнить ручной расчёт результатов и для контроля вывести их на печать, включая промежуточные.
- составить схему алгоритма и программу решения задачи.

Внимание! Выбор границ интервала $[X_0;X_k]$ осуществляется проверкой условия $X_0 < X_k$, что необходимо предусмотреть при составлении схемы и программы решения задачи. При этом значение X_0 должно быть существенно меньше X_k, а длина интервала $(X_k - X_0)$ достаточной, так как по условию задачи в матрице должны быть элементы, лежащие внутри (вне) интервала.

Примечание. При выборе чётной строки/столбца номер **i,j** изменяется в интервале [2,n], а нечётной – в интервале [1, n - 1] с шагом 2.

При решении задач используются алгоритмы вычисления суммы, произведения, количества, наибольшего (наименьшего) значения.

Задания для лабораторной работы 4.6

Определить:

№	Задача
1	Номера четного столбца с наибольшим количеством положительных и нечётного столбца с наименьшим количеством отрицательных элементов. Эти значения задают интервал $[X_0;X_k]$. Найти сумму элементов верхнего треугольника, лежащих вне интервала
2	Наибольший из элементов чётных и наименьший из элементов нечётных строк матрицы. Эти значения задают интервал $[X_0;X_k]$. Найти произведение элементов нижнего треугольника, лежащих внутри интервала

№	Задача
3	Номера строки наибольшего из элементов верхнего и столбца наименьшего из элементов нижнего треугольника. Эти значения задают интервал $[X_0;X_k]$. Найти сумму элементов главной диагонали, лежащих внутри интервала
4	Наименьший из элементов нечётных и наибольший из элементов четных столбцов матрицы. Эти значения задают интервал $[X_0;X_k]$. Найти количество положительных элементов верхнего треугольника, лежащих вне интервала
5	Номера строк с наибольшим и наименьшим из элементов главной диагонали. Эти значения задают интервал $[X_0;X_k]$. Найти количество элементов нижнего треугольника, лежащих внутри интервала
6	Количество положительных и отрицательных элементов главной диагонали. Эти значения задают интервал $[X_0;X_k]$. Найти наибольший из элементов верхнего треугольника, лежащих вне интервала
7	Наибольший и наименьший элементы главной диагонали. Эти значения задают интервал $[X_0;X_k]$. Найти сумму положительных элементов нижнего треугольника, лежащих внутри интервала
8	Наименьший из положительных элементов нижнего и наибольший из отрицательных элементов верхнего треугольника. Эти значения задают интервал $[X_0;X_k]$. Найти количество элементов главной диагонали, лежащих вне интервала
9	Наибольший элемент верхнего и наименьший элемент нижнего треугольника. Эти значения задают интервал $[X_0;X_k]$. Найти среднее арифметическое элементов матрицы, лежащих внутри интервала
10	Наименьший из элементов главной диагонали с нечётными номерами и наибольший из них с чётными номерами. Эти значения задают интервал $[X_0;X_k]$. Найти среднее арифметическое элементов верхнего треугольника, лежащих вне интервала

5. Справочные материалы

Существенную помощь в вычислениях с вещественными данными оказывают математические функции. Список некоторых наиболее употребительных функций приведён в таблице 5.1.

Примечание. На Fortran'е данные функции и их аргументы имеют тип **REAL,** на С – тип **double,** например для **ln x – double log (double x)**.

Названия стандартных функций в С следует набирать только строчными буквами!

Таблица 5.1

Стандартные функции C, Fortran, Basic

Функция		Запись на		
		C	Fortran	Basic
Натуральный логарифм	ln x	log(x)	LOG(X)	LOG(X)
Десятичный логарифм	lg x	log10(x)	LOG10(X)	
Квадратный корень	$\sqrt{x}$	sqrt(x)	SQRT(X)	SQR(X)
Абсолютная величина (модуль)	$\|x\|$	abs(x)	ABS(X)	ABS(X)
Экспонента	e^x	exp(x)	EXP(X)	EXP(X)
Тригонометрические функции (аргумент в радианах)				
Синус	sin x	sin(x)	SIN(X)	SIN(X)
Косинус	cos x	cos(x)	COS(X)	COS(X)
Тангенс	tg x	tan(x)	TAN(X)	TAN(X)
Обратные тригонометрические функции (аргумент $\|x\| \le 1$)				
Арксинус	arcsin x	asin(x)	ASIN(X)	
Арккосинус	arccos x	acos(x)	ACOS(X)	
Арктангенс	arctg(x)	atan(x)	ATAN(X)	ATN(X)
Гиперболические функции				
Синус	sh x	sinh(x)	SINH(X)	
Косинус	ch x	cosh(x)	COSH(X)	
Тангенс	th x	tanh(x)	TANH(X)	

Примечание. Гиперболические функции $\mathrm{sh}\,x$ и $\mathrm{ch}\,x$ выражаются через показательную функцию: $\mathrm{sh}x = \dfrac{e^x - e^{-x}}{2}$, $\mathrm{ch}x = \dfrac{e^x + e^{-x}}{2}$.

Пример

Запись арифметического выражения

$$1,5 \cdot ch^4 x + \sqrt[3]{|x|} \cdot e^{-x} - sin\sqrt{ln3} + tg3^\circ$$

на языках C, Fortran и Basic соответственно:

C: набирается строчными буквами

```
1.5*pow(cosh(x),4) + pow(abs(x),(1./3.))*exp(-x)-
sin(sqrt(log(3.)))+ tan(3*3.1415/180)
```

Fortran:

```
1.5*COSH(X)**4 + ABS(X)**(1./3.)*EXP(-X)-
SIN(SQRT(LOG(3.))) + TAN(3*3.1415/180)
```

Basic:

```
1.5*(EXP(X) + EXP(-X))^4 + ABS(X)^(1/3)*EXP(-X)-
SIN(SQR(LOG(3))) + TAN(3*3.1415/180)
```

Примеры записи выражений приведены также в работах 1; 4-6.

Отчёт-презентация к лабораторной работе 1

Вычисление значения выражения

1. Задание

Вычислить значение величины Z:

$$Z = (\ln|b| + \cos a)^2 - \frac{\sqrt{\ln 3 + \cos b}}{a\,(\ln a + \cos 2^\circ)}$$, где $a = 1{,}5 \quad b = -2.$

Для удобства написания программы введем обозначения:

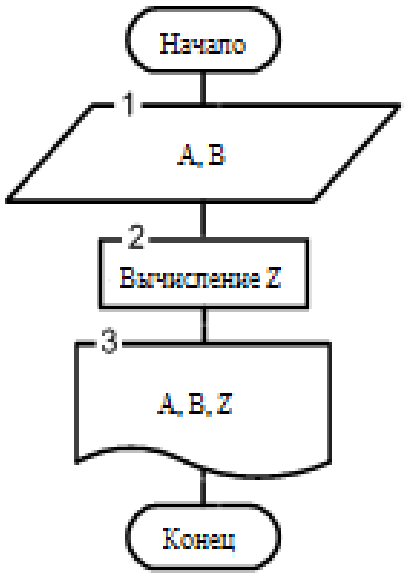
$$R = \ln|B| + \cos A; \quad W = \ln 3 + \cos B; \quad Q = \ln A + \cos 2^\circ.$$

В таком случае вычисление Z можно представить в виде:

$$Z = R^2 - \frac{\sqrt{W}}{A \cdot Q}$$

Обозначения переменных			
Имя переменной	Тип переменной		Пояснение
	Fortran	**C**	
A, B	**REAL**	**FLOAT**	Исходные данные
R, W, Q, Z	**REAL**	**FLOAT**	Рабочие переменные и результат

2. Схема алгоритма



3. Листинг Basic-программы

Программный код

```
' Исходные данные
A = 1.5: B = -2
PI = 3.1415
' Вычисление Z
R = LOG(ABS(B)) + COS(A)
W = LOG(3) + COS(B)
Q = LOG(A) + COS(2*PI/180)
Z = R^2 - SQR(W)/(A*Q)
' Вывод значений A,B,Z
PRINT " A = ";A;
PRINT " B = ";B
PRINT " Z = ";Z
END
```

4. Макет печати

```
A = 1.5 B = -2
Z = 0.1915
```

6. Ввод массива с датчиком случайных чисел

Ввод данных можно поручить датчику случайных чисел.

Задание

Сформировать матрицу A(n x n) и массивы X(n), Y(n), n=3. Для заполнения массива случайными числами необходимо задать диапазон [A,B], в границах которого будут находиться эти числа. На языке С выбор чисел осуществляется по формуле:

A + rand ()%(B - A)

Приведём программу ввода-вывода элементов массивов в интервале [–10;10] на языках С и Basic.

Листинг С-программы

Внимание. С *различает регистр букв, т.е., «**x**» и «**X**» – это две разные переменные. Нижняя граница индексов **i** и **j** задаётся равной нулю.*

Ввод-вывод матрицы A	Ввод-вывод массива X, Y

`#include<stdio.h>`
`#include<conio.h>`
`#include<stdlib.h>`
`#include<time.h>`
`void main()`
`{`
`int i, j, n = 3;`
`float a[3][3];`
`srand(time(0));`
`// Внешний цикл по i`
`for(i = 0; i<n; i++)`
`{`
`// Внутренний цикл по j`
`for(j = 0; j<n; j++)`
`{`
`a[i][j]=-10+rand()%20;`
`printf("%4.0f ",a[i][j]);`
`}`
`printf(" \n");`
`}`
`getch();`
`return;`
`}`
	`#include<stdio.h>`
	`#include<conio.h>`
	`#include<stdlib.h>`
	`#include<time.h>`
	`void main()`
	`{`
	`int i, n = 3;`
	`float x[3], y[3];`
	`srand(time(0));`
	`// Цикл по i`
	`for(i = 0; i<n; i++)`
	`{`
	`x[i]= -10 + rand()%20;`
	`y[i]= -10 + rand()%20;`
	`printf("%.of ",x[i]);`
	`printf("%.of \n",y[i]);`
	`}`
	`getch();`
	`return;`
	`}`

Результаты работы программы приведены в макете печати.

Задание чисел из диапазона [A;B] на языке Basic осуществляется по формуле (для целых чисел):

INT(A + (B - A) · RND)

Листинг Basic-программы

Программный код	
Ввод-вывод матрицы A	**Ввод-вывод массива X,Y**

`CLS`
`N = 3`
`RANDOMIZE TIMER`
`DIM A(N,N)`
`PRINT "МАТРИЦА A"`
`FOR I = 1 TO N`
` FOR J = 1 TO N`
`  A(I,J)=INT(-10 + 20*RND)`
`  PRINT USING "##",A(I,J);`
` NEXT J`
`PRINT ' Пропуск строки`
`NEXT I`
`END`
	`CLS`
	`N = 3`
	`RANDOMIZE TIMER`
	`DIM X(N),Y(N)`
	`PRINT TAB(3); " X  Y"`
	`FOR I = 1 TO N`
	` X(I)=INT(-10 + 20*RND)`
	` Y(I)=INT(-10 + 20*RND)`
	` PRINT USING "##",X(I);`
	` PRINT USING "##",Y(I)`
	`NEXT I`
	`END`

Макет печати результатов (по Basic-программе)

Матрица A		
–4	7	–5
8	6	–6
–7	–8	2

X	Y
8	–3
7	–6
2	8

Функция **srand(time(0))** и оператор **RANDOMIZE TIMER** нужны для того, чтобы каждый раз при выполнении программы создавалась новая последовательность псевдослучайных чисел.

Ручной расчёт результатов возможен только после выполнения программы, в которой необходимо заранее предусмотреть вывод искомого массива.

Библиографический список

1. **Артемов, И.Л.** Fortran: основы программирования / И.Л. Артемов. – М. : ДИАЛОГ-МИФИ, 2007. – 304 с.
2. **Бартеньев, О.В.** Современный Фортран / О.В. Бартеньев. – 4-е изд, доп. и перераб. – М. : ДИАЛОГ-МИФИ, 2005. – 560 с.
3. **Васильев А.Н.** Самоучитель C++ с примерами и задачами / А.Н. Васильев. – 2-е изд., перераб. – СПб. : Наука и техника, 2012. – 480 с.
4. **ГОСТ 19.701–90 (ИСО 5807–85).** Единая система программной документации. Схемы алгоритмов, программ, данных и систем. Обозначения условные и правила выполнения. – Взамен ГОСТ 19.002–80, ГОСТ 19.003–80 ; введ. 01.01.92. – М. : Изд-во стандартов, 1998. – 24 с.
5. **Гурьяшова, Р.Н.** Информатика : Сегментация программ : учеб. пособие для студ. оч. и заоч. обуч. технич. специальностей / Р.Н. Гурьяшова, А.В. Шеянов. – Н. Новгород : Изд-во ФГОУ ВПО «ВГАВТ», 2009. – 108 с.
6. **Гурьяшова, Р.Н.** Информатика. Теоретический курс : учеб. пособие для студ. всех форм обуч. спец-ти 280700.62 «Техносферная безопасность» / Р.Н. Гурьяшова, В.И. Логинов, Е.Ю.Седова. – Н. Новгород : Изд-во ФГОУ ВПО «ВГАВТ», 2013. – 84 с.
7. **Кетков, Ю.Л.** Практика программирования: Бейсик, Си, Паскаль / Ю.Л. Кетков, А.Ю. Кетков. – СПб. : БХВ, 2001. – 480 с.
8. **Культин, Н.Б.** C/C++ в задачах и примерах / Н.Б. Культин. – 2-е изд., перераб. и доп. – СПб. : БХВ-Петербург, 2011. – 368 с.
9. **Ляхович, В.Ф.** Основы информатики : учеб. / В.Ф. Ляхович, С.О. Крамаров. – 6-е изд., доп. и перераб. – Ростов н/Д : Феникс, 2007. – 713 с.
10. **Немнюгин, С.А.** Фортран в задачах и примерах / С.А. Немнюгин, О.Л. Стесик. – СПб. : БХВ-Петербург, 2008. – 320 с.
11. **Подбельский, В.В.** Программирование на языке Си / В.В. Подбельский, С.С. Фомин. – 2-е изд. – М. : Финансы и статистика, 1999. – 600 с.
12. **Рыжиков, Ю.И.** Современный Фортран : учеб. / Ю.И. Рыжиков. – СПб. : КОРОНА принт, 2007. – 288 с.

Оглавление

ВВЕДЕНИЕ ... 3

1. ПОРЯДОК ВЫПОЛНЕНИЯ ЛАБОРАТОРНОЙ РАБОТЫ ... 4

2. ПРИМЕРЫ ВЫПОЛНЕНИЯ ЛАБОРАТОРНЫХ РАБОТ ... 5

Лабораторная работа 1. Вычисление значения выражения ... 5
Лабораторная работа 2.
Вычисление значения величины, заданной в форме ветвления ... 8
Лабораторная работа 3.
Вычисление значения величины, заданной в форме ветвления ... 11

КОНТРОЛЬНЫЕ ЗАДАНИЯ ДЛЯ ПРОВЕРКИ ЗНАНИЙ ... 15

Лабораторная работа 4. Исследование функции одной переменной ... 17
Лабораторная работа 5.
Исследование функции одной переменной, зависящей от параметра ... 23
Лабораторная работа 6. Исследование функции одной переменной с различным шагом варьирования аргумента ... 29

КОНТРОЛЬНОЕ ЗАДАНИЕ ДЛЯ ПРОВЕРКИ ЗНАНИЙ. РАСЧЁТ ОРДИНАТ ЭПЮРЫ ДАВЛЕНИЯ ГРУНТА НА ПОДПОРНЫЕ СТЕНКИ ... 35

Лабораторная работа 7. Исследование функции двух переменных ... 36
Лабораторная работа 8. Исследование функции двух переменных при фиксированном значении аргумента ... 41
Лабораторная работа 9. Вычисление координат искомой точки (наиболее удалённой от начала координат / ближайшей к нему) ... 45
Лабораторная работа 10. Вычисление координат искомой точки (ближайшей к заданной / наиболее удалённой от неё) ... 50
Лабораторная работа 11. Вычисление координат искомой точки ... 54
Лабораторная работа 12. Вычисление значения выражения ... 58
Лабораторная работа 13. Анализ эффективности работы предприятия ... 61
Лабораторная работа 14. Анализ итогов сдачи сессии ... 66

КОНТРОЛЬНОЕ ЗАДАНИЕ ДЛЯ ПРОВЕРКИ ЗНАНИЙ. ВЫЧИСЛЕНИЕ ПЛОЩАДИ ШПАНГОУТА ... 70

Лабораторная работа 15. Анализ эффективности работы танкерного флота 72
Лабораторная работа 16. Анализ итогов сессии в группе ... 83
Лабораторная работа 17. Формирование одномерного массива ... 87

КОНТРОЛЬНОЕ ЗАДАНИЕ ДЛЯ ПРОВЕРКИ ЗНАНИЙ ... 92

3. ЗАДАНИЯ НА ВЫПОЛНЕНИЕ ЛАБОРАТОРНЫХ РАБОТ.....94

Таблица 3.1 Вычисление значения выражения.....94
Таблица 3.2
Вычисление значения величины, заданной в форме ветвления 96
Таблица 3.3
Вычисление значения величины, заданной в форме ветвления 99
Таблица 3.4 Исследование функции одной переменной 100
Таблица 3.5
Исследование функции одной переменной, зависящей от параметра 104
Таблица 3.6 Исследование функции одной переменной с различным шагом варьирования аргумента 107
Таблица 3.7 Исследование функции двух переменных 109
Таблица 3.8 Исследование функции двух переменных при фиксированном значении аргумента.....111
Таблица 3.9 Вычисление координат точки, наиболее удалённой от начала координат (ближайшей к нему).....113
Таблица 3.10 Вычисление координат искомой точки, ближайшей к заданной (наиболее удалённой от неё) 115
Таблица 3.11 Вычисление координат искомой точки 116
Таблица 3.12 Вычисление значения выражения 117
Таблица 3.13 Анализ эффективности работы предприятия 119
Таблица 3.14 Анализ итогов экзамена в группе 121
Таблица 3.15 Анализ эффективности работы флота 122
Таблица 3.16 Анализ итогов сессии в группе.....124
Таблица 3.17 Формирование одномерного массива 125

4. ЗАДАНИЯ ДЛЯ САМОСТОЯТЕЛЬНОЙ РАБОТЫ.....127

Лабораторная работа 4.1. Вычисление значения функции.....127
Лабораторная работа 4.2.
Вычисление функции, заданной в виде ветвления.....128
Лабораторная работа 4.3. Расчёт грузовой шкалы судна 130
Лабораторная работа 4.4. Вычисление координат точки 132
Лабораторная работа 4.5.
Вычисление площади живого сечения однорукавного русла.....134
Лабораторная работа 4.6. Вычисление границ искомого интервала 134

5. СПРАВОЧНЫЕ МАТЕРИАЛЫ.....136

Таблица 5.1. Стандартные функции C, Fortran, Basic 138
Отчёт-презентация к лабораторной работе 1.....139

6. ВВОД МАССИВА С ДАТЧИКОМ СЛУЧАЙНЫХ ЧИСЕЛ 136

БИБЛИОГРАФИЧЕСКИЙ СПИСОК 140

Гурьяшова Римма Николаевна

Лабораторный практикум по информатике

Учебное пособие

Отпечатано без издательского редактирования и корректирования

Подписано в печать 16.03.15.
Формат 60×84 $^{1}/_{16}$. Гарнитура «Таймс».
Ризография. Усл. печ. л. 8,6. Тираж 80 экз. Заказ 190.

Издательско-полиграфический комплекс ФГБОУ ВО «ВГУВТ»
603950, Нижний Новгород, ул. Нестерова, 5

145

Требования к содержанию и оформлению отчета
по лабораторным работам
по дисциплине «Информатика»
за первый семестр

(специальность 26.05.07 «Эксплуатация судового электрооборудования и средств автоматики»,
профиль «Эксплуатация судового электрооборудования и средств автоматики»)

Цель дисциплины – сформировать у обучающихся следующие компетенции:

ОПК-5: Способен понимать принципы работы современных информационных технологий и использовать их для решения задач профессиональной деятельности;
ПК-6: Способен осуществлять безопасное техническое использование, техническое обслуживание судовой компьютерной информационной системы в соответствии с международными и национальными требованиями;

Отчет должен содержать:
1) Стандартным образом оформленный титульный лист.

2) К каждой лабораторной работе:
а) формулировку задания;
б) блок- схему алгоритма;
в) листинг программы;
г) скриншот результатов работы программы.

Примерный перечень лабораторных работ:

1. Разветвленные вычислительные процессы (задача 1, задача 2, выполненные на языке Си).
2. Циклические вычислительные процессы. Исследование функции одной переменной (табулирование функции одной переменной).
3. Циклические вычислительные процессы. Обработка данных одномерных массивов (задача 1, задача 2, выполненные на языке Си).
4. Циклические вычислительные процессы. Обработка данные двумерных массивов (задача 1, задача 2, выполненные на языке Си).
5. Программирование пользовательских функций.

Примечание. Все листы отчета должны быть оформлены в простую рамку.

Внимание!
1. Отчет по лабораторным работам должен быть готов ко дню проведения зачетного занятия!
2. Перед тем как распечатать отчет, продемонстрируйте его преподавателю в электронном виде, а затем устраните замечания!

МИНТРАНС РОССИИ
РОСМОРРЕЧФЛОТ

Федеральное государственное бюджетное образовательное учреждение высшего образования

ВОЛЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ВОДНОГО ТРАНСПОРТА
(ФГБОУ ВО «ВГУВТ»)

ул. Нестерова, 5а, Нижний Новгород, 603951
Тел. (831) 419-47-56

Кафедра «Систем информационной безопасности, управления и телекоммуникаций»
20 _-20___учебный год

Экзамен по дисциплине «Информатика»
Специальность 26.05.07
«Эксплуатация судового электрооборудования и средств автоматики»
Специализация Эксплуатация судового электрооборудования и средств автоматики

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ №1

1. Информационно-телекоммуникационные технологии глобальной компьютерной сети Интернет.
2. Ввод-вывод данных в Си. Форматный ввод-вывод.

Зав. кафедрой, д.т.н., профессор Федосенко Ю.С.

МИНТРАНС РОССИИ
РОСМОРРЕЧФЛОТ

Федеральное государственное бюджетное образовательное учреждение высшего образования

ВОЛЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ВОДНОГО ТРАНСПОРТА
(ФГБОУ ВО «ВГУВТ»)

ул. Нестерова, 5а, Нижний Новгород, 603951
Тел. (831) 419-47-56

Кафедра «Систем информационной безопасности, управления и телекоммуникаций»
20 _-20___учебный год

Экзамен по дисциплине «Информатика»
Специальность 26.05.07
«Эксплуатация судового электрооборудования и средств автоматики»
Специализация Эксплуатация судового электрооборудования и средств автоматики

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ №2

1. Структура и принципы функционирования глобальной сети Интернет.
2. Различие междупостфиксной и префиксной операцией инкремента(декремента).

Зав. кафедрой, д.т.н., профессор Федосенко Ю.С.

МИНТРАНС РОССИИ
РОСМОРРЕЧФЛОТ

Федеральное государственное бюджетное образовательное учреждение высшего образования

ВОЛЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ВОДНОГО ТРАНСПОРТА
(ФГБОУ ВО «ВГУВТ»)

ул. Нестерова, 5а, Нижний Новгород, 603951
Тел. (831) 419-47-56

Кафедра «Систем информационной безопасности, управления и телекоммуникаций»
20 _-20___учебный год

Экзамен по дисциплине «Информатика»
Специальность 26.05.07
«Эксплуатация судового электрооборудования и средств автоматики»
Специализация Эксплуатация судового электрооборудования и средств автоматики

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ №3

1. Основы технологии искусственного интеллекта.
2. Операции присваивания в С++.

Зав. кафедрой, д.т.н., профессор Федосенко Ю.С.

МИНТРАНС РОССИИ
РОСМОРРЕЧФЛОТ

Федеральное государственное бюджетное образовательное учреждение высшего образования

ВОЛЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ВОДНОГО ТРАНСПОРТА
(ФГБОУ ВО «ВГУВТ»)

ул. Нестерова, 5а, Нижний Новгород, 603951
Тел. (831) 419-47-56

Кафедра «Систем информационной безопасности, управления и телекоммуникаций»
20 _-20___учебный год

Экзамен по дисциплине «Информатика»
Специальность 26.05.07
«Эксплуатация судового электрооборудования и средств автоматики»
Специализация Эксплуатация судового электрооборудования и средств автоматики

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ №4

1. Адресация в сетях.
2. Организация подпрограмм-функций. Рекурсия.

Зав. кафедрой, д.т.н., профессор Федосенко Ю.С.

МИНТРАНС РОССИИ
РОСМОРРЕЧФЛОТ

Федеральное государственное бюджетное образовательное учреждение высшего образования

ВОЛЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ВОДНОГО ТРАНСПОРТА
(ФГБОУ ВО «ВГУВТ»)

ул. Нестерова, 5а, Нижний Новгород, 603951
Тел. (831) 419-47-56

Кафедра «Систем информационной безопасности, управления и телекоммуникаций»
20 _-20___учебный год

Экзамен по дисциплине «Информатика»
Специальность 26.05.07
«Эксплуатация судового электрооборудования и средств автоматики»
Специализация Эксплуатация судового электрооборудования и средств автоматики

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ №5

1. Технологии передачи данных. Протоколы.
2. Условный оператор. Примеры.

Зав. кафедрой, д.т.н., профессор Федосенко Ю.С.

МИНТРАНС РОССИИ
РОСМОРРЕЧФЛОТ

Федеральное государственное бюджетное образовательное учреждение высшего образования

ВОЛЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ВОДНОГО ТРАНСПОРТА
(ФГБОУ ВО «ВГУВТ»)

ул. Нестерова, 5а, Нижний Новгород, 603951
Тел. (831) 419-47-56

Кафедра «Систем информационной безопасности, управления и телекоммуникаций»
20 _-20___учебный год

Экзамен по дисциплине «Информатика»
Специальность 26.05.07
«Эксплуатация судового электрооборудования и средств автоматики»
Специализация Эксплуатация судового электрооборудования и средств автоматики

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ №6

1. Информационные технологии программирования на примере языка Си в среде MSVisual Studio 2010.
2. Оператор множественного выбора. Примеры.

Зав. кафедрой, д.т.н., профессор Федосенко Ю.С.

МИНТРАНС РОССИИ
РОСМОРРЕЧФЛОТ

Федеральное государственное бюджетное образовательное учреждение высшего образования

ВОЛЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ВОДНОГО ТРАНСПОРТА
(ФГБОУ ВО «ВГУВТ»)

ул. Нестерова, 5а, Нижний Новгород, 603951
Тел. (831) 419-47-56

Кафедра «Систем информационной безопасности, управления и телекоммуникаций»
20 _-20___учебный год

Экзамен по дисциплине «Информатика»
Специальность 26.05.07
«Эксплуатация судового электрооборудования и средств автоматики»
Специализация Эксплуатация судового электрооборудования и средств автоматики

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ №7

1. Алгоритмы. Типы алгоритмов. Способы записи алгоритмов.
2. Оператор цикла for. Примеры.

Зав. кафедрой, д.т.н., профессор Федосенко Ю.С.

МИНТРАНС РОССИИ
РОСМОРРЕЧФЛОТ

Федеральное государственное бюджетное образовательное учреждение высшего образования

ВОЛЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ВОДНОГО ТРАНСПОРТА
(ФГБОУ ВО «ВГУВТ»)

ул. Нестерова, 5а, Нижний Новгород, 603951
Тел. (831) 419-47-56

Кафедра «Систем информационной безопасности, управления и телекоммуникаций»
20 _-20___учебный год

Экзамен по дисциплине «Информатика»
Специальность 26.05.07
«Эксплуатация судового электрооборудования и средств автоматики»
Специализация Эксплуатация судового электрооборудования и средств автоматики

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ №8

1. Типовые алгоритмические конструкции. Алгоритмы вычисления суммы,произведения, количества, минимума и максимума.
2. Оператор цикла while. Примеры.

Зав. кафедрой, д.т.н., профессор Федосенко Ю.С.

МИНТРАНС РОССИИ
РОСМОРРЕЧФЛОТ

Федеральное государственное бюджетное образовательное учреждение высшего образования

ВОЛЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ВОДНОГО ТРАНСПОРТА
(ФГБОУ ВО «ВГУВТ»)

ул. Нестерова, 5а, Нижний Новгород, 603951
Тел. (831) 419-47-56

Кафедра «Систем информационной безопасности, управления и телекоммуникаций»
20 _-20___учебный год

Экзамен по дисциплине «Информатика»
Специальность 26.05.07
«Эксплуатация судового электрооборудования и средств автоматики»
Специализация Эксплуатация судового электрооборудования и средств автоматики

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ №9

1. Структура программы на языке Си.
2. Оператор цикла do while. Примеры.

Зав. кафедрой, д.т.н., профессор Федосенко Ю.С.

МИНТРАНС РОССИИ
РОСМОРРЕЧФЛОТ

Федеральное государственное бюджетное образовательное учреждение высшего образования

ВОЛЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ВОДНОГО ТРАНСПОРТА
(ФГБОУ ВО «ВГУВТ»)

ул. Нестерова, 5а, Нижний Новгород, 603951
Тел. (831) 419-47-56

Кафедра «Систем информационной безопасности, управления и телекоммуникаций»
20 _-20___учебный год

Экзамен по дисциплине «Информатика»
Специальность 26.05.07
«Эксплуатация судового электрооборудования и средств автоматики»
Специализация Эксплуатация судового электрооборудования и средств автоматики

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ №10

1. Что такое тип данных? Простые типы данных. Описание данных. Приведение и преобразование типов.
2. Особенности использования операторов цикла. Бесконечные циклы.

Зав. кафедрой, д.т.н., профессор Федосенко Ю.С.

МИНТРАНС РОССИИ
РОСМОРРЕЧФЛОТ

Федеральное государственное бюджетное образовательное учреждение высшего образования

ВОЛЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ВОДНОГО ТРАНСПОРТА
(ФГБОУ ВО «ВГУВТ»)

ул. Нестерова, 5а, Нижний Новгород, 603951
Тел. (831) 419-47-56

Кафедра «Систем информационной безопасности, управления и телекоммуникаций»
20 _-20___учебный год

Экзамен по дисциплине «Информатика»
Специальность 26.05.07
«Эксплуатация судового электрооборудования и средств автоматики»
Специализация Эксплуатация судового электрооборудования и средств автоматики

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ №11

1. Что такое переменная? Чем объявление переменной отличается от ее определения? Привести примеры определений и объявлений.
2. Что такое массив? Обработка данных массива.

Зав. кафедрой, д.т.н., профессор Федосенко Ю.С.

МИНТРАНС РОССИИ
РОСМОРРЕЧФЛОТ

Федеральное государственное бюджетное образовательное учреждение высшего образования

ВОЛЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ВОДНОГО ТРАНСПОРТА
(ФГБОУ ВО «ВГУВТ»)

ул. Нестерова, 5а, Нижний Новгород, 603951
Тел. (831) 419-47-56

Кафедра «Систем информационной безопасности, управления и телекоммуникаций»
20 _-20___учебный год

Экзамен по дисциплине «Информатика»
Специальность 26.05.07
«Эксплуатация судового электрооборудования и средств автоматики»
Специализация Эксплуатация судового электрооборудования и средств автоматики

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ №12

1. Выражение в языке Си. Составные части выражения. Операнды.
2. Виды массивов и их инициализация. Обработка массивов.

Зав. кафедрой, д.т.н., профессор Федосенко Ю.С.

МИНТРАНС РОССИИ
РОСМОРРЕЧФЛОТ

Федеральное государственное бюджетное образовательное учреждение высшего образования

ВОЛЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ВОДНОГО ТРАНСПОРТА
(ФГБОУ ВО «ВГУВТ»)

ул. Нестерова, 5а, Нижний Новгород, 603951
Тел. (831) 419-47-56

Кафедра «Систем информационной безопасности, управления и телекоммуникаций»
20 __-20____учебный год

Экзамен по дисциплине «Информатика»
Специальность 26.05.07
«Эксплуатация судового электрооборудования и средств автоматики»
Специализация Эксплуатация судового электрооборудования и средств автоматики

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ №13

1. Что такое отношение? В каком случае оно считается ложным, а в каком –истинным?
2. Объявление и определение функции.

Зав. кафедрой, д.т.н., профессор Федосенко Ю.С.

МИНТРАНС РОССИИ
РОСМОРРЕЧФЛОТ

Федеральное государственное бюджетное образовательное учреждение высшего образования

ВОЛЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ВОДНОГО ТРАНСПОРТА
(ФГБОУ ВО «ВГУВТ»)

ул. Нестерова, 5а, Нижний Новгород, 603951
Тел. (831) 419-47-56

Кафедра «Систем информационной безопасности, управления и телекоммуникаций»
20 _-20___учебный год

Экзамен по дисциплине «Информатика»
Специальность 26.05.07
«Эксплуатация судового электрооборудования и средств автоматики»
Специализация Эксплуатация судового электрооборудования и средств автоматики

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ №14

1. Какие операции называются унарными? Привести примеры.
2. Логические операции. Отличие «одинарных» операций от «двойных».

Зав. кафедрой, д.т.н., профессор Федосенко Ю.С.

МИНТРАНС РОССИИ
РОСМОРРЕЧФЛОТ

Федеральное государственное бюджетное образовательное учреждение высшего образования

ВОЛЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ВОДНОГО ТРАНСПОРТА
(ФГБОУ ВО «ВГУВТ»)

ул. Нестерова, 5а, Нижний Новгород, 603951
Тел. (831) 419-47-56

Кафедра «Систем информационной безопасности, управления и телекоммуникаций»
20 _-20___учебный год

Экзамен по дисциплине «Информатика»
Специальность 26.05.07
«Эксплуатация судового электрооборудования и средств автоматики»
Специализация Эксплуатация судового электрооборудования и средств автоматики

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ №15

1. Какие операции называются бинарными? Привести примеры.
2. Этапы создания исполняемой программы на языке Си.

Зав. кафедрой, д.т.н., профессор  Федосенко Ю.С.

Задания по дисциплине «Информатика»
к разделу «Программирование микропроцессоров»
(специальность 26.05.07 «Эксплуатация судового электрооборудования и средств автоматики»,
профиль «Эксплуатация судового электрооборудования и средств автоматики»)

Цель дисциплины – сформировать у обучающихся следующие компетенции:

ОПК-5: Способен понимать принципы работы современных информационных технологий и использовать их для решения задач профессиональной деятельности;

ПК-6: Способен осуществлять безопасное техническое использование, техническое обслуживание судовой компьютерной информационной системы в соответствии с международными и национальными требованиями;

КРАТКАЯ ХАРАКТЕРИСТИКА МИКРОПРОЦЕССОРОВ МП К580

Длина слова МП К 580- 8 битов или байт. При обращении к памяти используются 16-битные адреса, что позволят обращаться к 65536 ячейкам памяти.

МП имеет 8-битный аккумулятор (регистр A), шесть 8-битных регистров общего назначения (B, C, D, E, H, L), а также регистр флагов F. Регистры общего назначения могут объединяться в 16-битные регистры двойной длины.

Содержимое аккумулятора и регистра флагов объединяется в «слово состояния процессора» (PSW).

В МП результат характеризуется следующими флагами:

(1- флаг установлен, 0-сброшен):

флаг знака S (SING); совпадает со старшим битом результата;

флаг нуля Z (ZERO); устанавливается при нулевом результате;

флаг вспомогательного переноса AC(AUXILIARY CARRY); устанавливается при возникновении переноса из третьего бита в четвертый;

флаг паритета P (PARITY); устанавливается при наличии в результате четного числа единиц;

флаг переноса CA (CARRY); устанавливается при переносе из старшего бита результата (при вычитании становится флагом заема);

Команды МП по своим форматам делятся на одно-, двух- и трехбайтные.

При написании программ используются сокращенные английские наименования команд (см. приложение).

Для успешной подготовки программ необходимо представлять совокупность элементов УОУ, их назначение и взаимодействие, т.е. программистскую модель УОУ. Такая модель изображена на рис. 1.

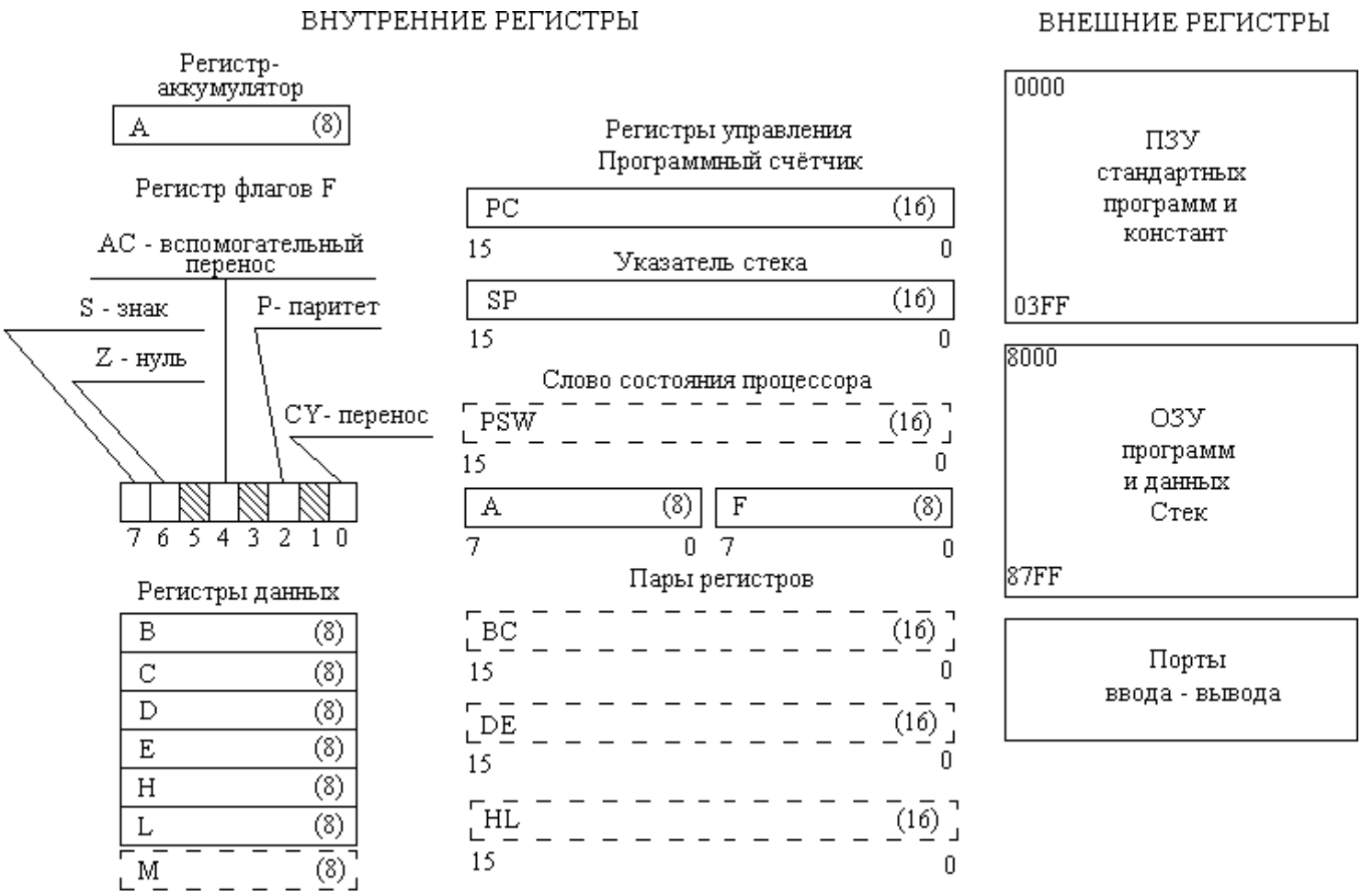


Рис. 1

1. СИСТЕМА КОМАНД

1.1 Символы и обозначения, используемые в командах МП.

Байт- 8- битное слово данных;

PC (PROGRAM COUNT)- 16- битный программный счетчик

(PC_H и PC_l - старший и младший байты счетчика);

SP (STACK POINTER)- 16- битный указатель стека

(SP_H и SP_l - старший и младший байты указателя);

SГ- 16- битная вершина стека;

PSW (PROCESSOR STATUS WORD)- слово состояния

процессора;

A- регистр- аккумулятор;

r- один из регистров: B, C, D, E, H, L ;

r r- один из двойных регистров: BC, DE, HL, SP,

PSW ($rr_н$ и $rr_л$ - старший и младший байты двойного регистра) ;

M (MEMORY)- ячейка памяти, адрес которой хранится в паре

регистров HL ;

v- 8-битное значение (байт данных);

vv-16- битное значение;

aa- 16- битный адрес;

S- флаг знака;

Z- флаг нуля;

P- флаг паритета;

AC- флаг вспомогательного переноса;

CY-флаг переноса;

< - операция отсылки;

- - операция обмена;

( )-в скобки заключается содержимое регистра или ячейки памяти.

2.1. Команды пересылки

Важно!!! Команды пересылки состояния флагов не изменяют.

Операция	Мнемоника	Кодкоманды	Действие
1	2	3	4

Формат команд
1 байт
Пересылка данных
	MOV r1, r2
	MOV r, M
	MOV M, r
		см.
		ниже
		ОА
			(r1) < (r2)
			(r) < (M)
			(M) < ( )

Косвенная загрузка аккумулятора	LDAX BC	1A	(A) < ((BC))
Косвенная запись аккумулятора	LDAX DE		(A) < ((DE))
		O2	((BC)) < (A)
Обмен	STAX BC	12	((DE)) < (A)
Формат команд-	STAX DE		
2 байта		EB	(HL) < (DE)
непосредствен-	XCHG	см.	(r) < *v*
ная пересылка	MVI, r, *v*	ниже	(r) < *v*
Формат команд	MVI M, *v*		
3 байта	LXI BC, *vv*	O1	(BC) < *vv*
непосредсвтен-	LXI DE, *vv*	11	(DE) < *vv*
ная загрузка	LXI HL, *vv*	21	(HL) < *vv*
Прямая загрузка	LXI SP, *vv*	31	(SP) < *vv*
аккумулятора	LDA *aa*	3A	
Прямая запись аккумулятора	STA *aa*		(A) < *(aa)*
Загрузка пары регистровHL	LHLD *aa*	32	
Запись содержимого HL в память	SHLD *aa*	2A	*(aa)* < (A)
		22	(L) < (*aa*)
			(H) < (*a*+ 1)
			(*aa*) < (L)
			(*aa*+1) < (H)

Коды команд пересылки

Регистр-приемник	Пересылка данных между регистрами								Непосредственная пересылка MVI r MVI M
	Регистр- источник								
	B	C	D	E	H	L	M	A	
B	40	41	42	43	44	45	46	47	06
C	48	49	4A	4B	4C	4D	4E	4F	0E
D	50	51	52	53	54	55	56	57	16
E	58	59	5A	5B	5C	5D	5E	5F	1E
H	60	61	62	63	64	65	66	67	26
L	68	69	6A	6B	6C	6D	6E	6F	2E
M	70	71	72	73	74	75	-	77	36
A	78	79	7A	7B	7C	7D	7E	7F	3E

1.2 Арифметические команды

Операция	Мнемоника	Действие	Флаги
1	2	3	4
Формат команд–2 байта	ADI *v*	(A)<(A)+*v*	Все
Непосредственное сложение	ACI *v*	(A)<(A)+*v*+(CY)	Все
Непосредственное сложение с	SUI *v*	(A)<(A)-*v*	
переносом	SBI *v*	(A)<(A)-*v*-(CY)	
Непосредственное вычитание	ADD r	(A)<(A)+(r)	
Непосредственное вычитание	ADD *м*	(A)<(A)+(*м*)	
с заемом	ADC *r*	(A)<(A)+(r)+(CY)	

Формат команд – 2 байта	ADC *м*	(A)<(A)+(*м*)+(CY)	
Сложение	SUB *r*	(A)<(A)-(r)	
Сложение с переносом	SUB *м*	(A)<(A)-(*м*)	
Вычитание	SBB *r*	(A)<(A)-(r)-(CY)	
Вычитание с заемом	SBB *м*	(A)<(A)-(*м*)-(CY)	
Увеличение	INR *r*	(r)<(r)+1	Z,S
Уменьшение	INR *м*	(*м*)<(*м)*+1	P,AC
Сложение регистровых пар	INX *r r*	(rr)<(rr)+1	Нет
Десятичное дополнение аккумулятора	DCR *v*	(*v*)<(*v*)+1	Z,S
	DCR *м*	(*м*)<(*м)*+1	P,AC
	DCX *r r*	(*r r*)<(*r r*)+1	Нет
	DAD *r r*	(HL)<(HL)+(rr)	CY
	DAR *r r*	(A) приводится к BCD [x)]	Все

[x)]BCD(BINARY CODED DECIMAL)–формат двоично-кодированых десятичных чисел. При котором байт содержит две десятичные цифры (тетрады). Команда DAA(ее код–27)в два приема корректирует 8–битное значение аккумулятора:1).Если младшая тетрада больше 9 или установлен флаг вспомогательного переноса AC,то содержимое аккумулятора увеличивается на 6;2).Если после того старшая тетрада больше 9 или установлен флаг CY,то старшая тетрада увеличивается на 6.

Коды арифметических операций с аккумулятором

Операция	Операция с регистром *r* или ячейкой *M*									Операция с байтом данных	
	Мнемоника	Регистр-источник(…)								Мнемоника	Код
		B	C	D	E	H	L	M	A		
Сложение	ADD…	80	81	82	83	84	85	86	87	ADI *v*	C6

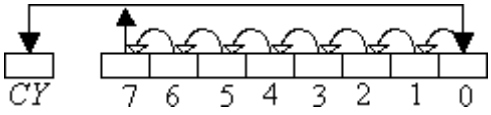
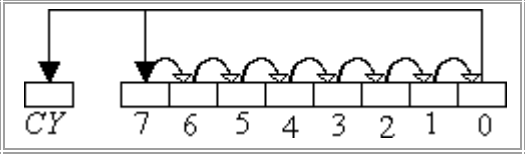
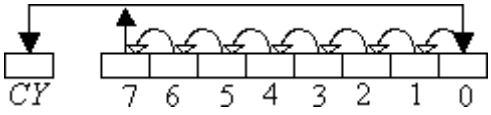
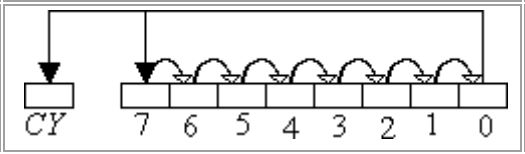
Сложение с переносом	ADC…	88	89	8A	8B	8C	8D	8E	8F	ACI *v*	CE
Вычитание	SUB…	90	91	92	93	94	95	96	97	SUI *v*	D6
Вычитание с заемом	SBB…	98	99	9A	9B	9C	9D	9E	9F	SBI *v*	DE
Увеличение	INR…	04	0C	14	1C	24	2C	34	3C	-	-
Уменьшение	DCR…	05	0D	15	1D	25	2D	35	3D	-	-

Коды арифметических операций с двойными регистрами

Операции	Мнемоника	Двойной регистр *rr*(…)			
		BC	DE	HL	SD
Увеличение	INX…	03	13	23	33
Уменьшение	DCX…	0B	1B	2B	3B
Сложение регистровых пар	DAD…	09	19	29	39

2.3. Команда логических операций и сдвига

Операция (ее код)	Мнемоника	Действие	Флаги
Формат команда-2 байта			
Логическое ”И”	ANI v	(A)←(A) ^*r*	Z,S,P;AC=A(3);CY=0
Исключающее “ИЛИ”	XRI v	(A)←(A)∀*v*	Z,S,P;AC=CY=0
Логическое ”ИЛИ”	ORI v	(A)←(A) V *v*	
Сравнение	CPI v	(A) - *v*	Всё
Формат команда-1 байта			
Логическое ”И” (поразрядная коньюкция)	ANA r AMA M	(A)←(A) ^ (r) (A)←(A) ^ (M)	Z,S,P;AC=A(3);CY=0
Исключающее	XRA r	(A)←(A) ∀(r)	Z,S,P;AC=CY=0

“ИЛИ” (поразрядное сложение по mod 2)	XRA M	(A)←(A) ∀(M)	
Логическое ”ИЛИ” (поразрядная дизъюкция)	ORA r ORA M	(A)←(A) V(r) (A)←(A) V (M)	
Сравнение (без записи результата)	CMP r CMP M	(A)-(r) (A)-(M)	Всё
Инверсия аккумулятора	CMA	(A) ← (A)	Нет
Инверсия переноса	CMC	(CY) ← (CY)	CY
Установка переноса	STC	(CY) ←1	
Циклический сдвиг влево (07)	RLC		CY=A (7)
Циклический сдвиг вправо (0f)	RRC		CY=A (0)
Арифметический сдвиг влево (17)	RAL		CY=A (7)
Арифметический сдвиг вправо (1F)	RAR		CY=A (0)

Коды логических операций

Операция	Операция с регистром *r* или ячейкой *M*									Операция с байтом данных	
	Мнемоника	Регистр-источник(…)								Мнемоника	Код
		B	C	D	E	H	L	M	A		
Логическое ”И”	ANA …	A0	A1	A2	A3	A4	A5	A6	A7	ANI *v*	E6
Исключающее	XRA …	A8	A9	AA	AB	AC	AD	AE	AF	XRI *v*	EE

“ИЛИ”											
Логическое ”ИЛИ”	ORA …	B0	B1	B2	B3	B4	B5	B6	B7	ORI *v*	F6
Сравнение	CMP …	B8	B9	BA	BB	BC	BD	BE	BF	CPI *v*	FE

2.4. Команды передачи управления и обращения к подпрограммам

Команды этой группы не изменяют состояние флагов, но могут использовать флаги Z, CY, P, S. Именно так команды перехода, вызова и возврата по условию проверяют по 4 пары дополняющих друг друга условий.
Обозначим (COND)- одно из 8 условий:
NZ-ненулевой результат (Z=0);
Z–нулевой результат (Z=1);
NC–отсутствие переноса из старшего разряда или заёма (CY=0);
C–наличие переноса или заёма (CY=1);
PO-нечётность числа единиц в результате (P=0);
PE -чётность числа единиц в результате (P=1);
P–“плюс” (S=0);
M–“минус” (S=1).
Команды условного и безусловного перехода и вызова содержат адрес aa и поэтому имеют трёхбайтный формат вида:

Код команды
Младший полуадрес
Старший полуадрес

Для перехода к подпрограммам и возврата из подпрограмм в МП К580 используется стек. Стек-группа последовательно пронумерованных ячеек памяти. Указатель стека (SP) постоянно хранит автоматически устанавливаемый адрес вершины стека (SP)- последней 13 занятой ячейки. 8-битное слово при записи его в стек помещается в очередную свободную ячейку стека, а при считывании из стека извлекается слово, поступившее в стек последним. Таким образом, стек автоматически реализует правило “последним пришел - первым ушел” (рис.2).

В МП К580 УОУ стек организован в виде группы ячеек, начинающийся с адреса 83E016 и расширяется по мере заполнения в сторону меньших адресов, т.е. при передаче слова памяти в стек значение SP уменьшается, а при извлечении слова из стека – увеличивается.

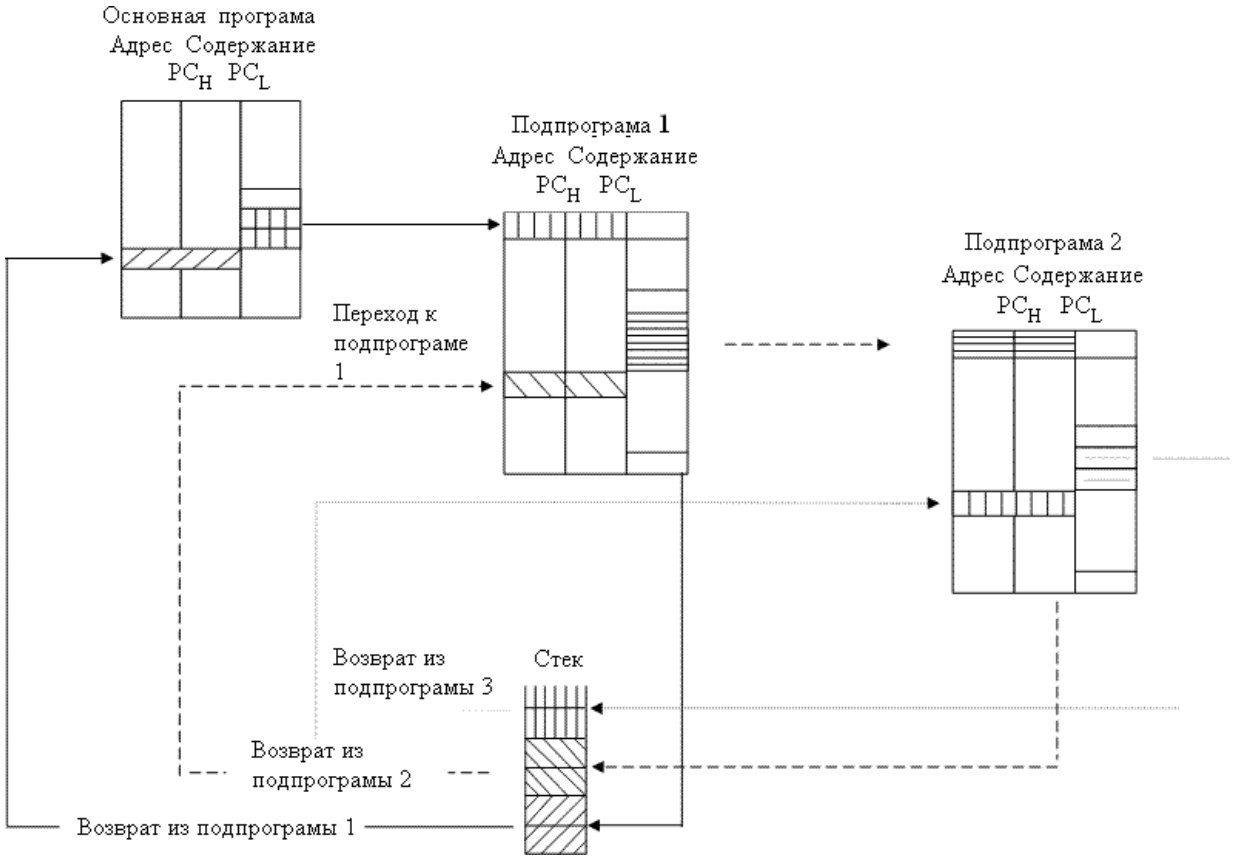
Взаимодействие указателя стека и программного счетчика (PC) при передачах управления показано ниже.

При использование стека, глубина вложения подпрограмм зависит только от выделенной стеку памяти и практически не ограниченна.

Рестарт используется для обработки прерываний и введения контрольных точек при отладке программ.

При рестарте (повторный запуск) текущее значение PC запоминается в стеке, а МП переходит к программе, начинающейся в ячейке с адресом вида 8N(0<N<7): 000016,000816,001016,…003816.

Особенной командой безусловной передачи управления без возврата является команда PCHL (её код – E9), по которой МП продолжает программу с адреса, хранящегося в паре регистров HL.



2. Упрощенная схема использования стека при обращениях к подпрограмме.

Операции	Мнемоника	Действие
Переход	*JVP aa*	*(PC) ← aa*
Переход по условию	*J(COND) aa*	Если условие верно, то выполняется команда *JMP aa*
Вызов	*CALL aa*	*((SP)-1) ← (PC_H), ((SP)-2) ←(PC_L),* *(SP) ←(SP)-2, (PC) ← aa*
Вызов по условию	*C(COND) aa*	Если условие верно, то выполняется команда *CALL aa*
Возврат	*RET*	*(PC_L) ← ((SP)), (PC_H) ← ((SP)+1),*

		(SP) ← (SP)+2
Возврат по условию	*R(COND)*	Если условие верно, то выполняется команда *RET*
Рестарт (0<*N*<7)	*RST N*	*((SP-1) ← (PC_H), ((SP)-2) ← (PC_L),* *(SP) ← (SP)-2, (PC) ← 8*N*
Передача из *HL* в *PC*	*PCHL*	*(PC) ← (HL)*

Коды команд управления

Операция	Мнемоника	Безусловный переход	Переход по условию(*COND*)							
			NZ	*Z*	*NC*	*C*	*PO*	*PE*	*P*	*M*
Вызов	CALL	CD	C4	CC	D4	DC	E4	EC	F4	FC
Переход	JMP	C3	C2	CA	D2	DA	E2	EA	F2	FA
Возврат	RET	C9	C0	C8	D0	D8	E0	E8	F0	F8

Коды рестарта

N	0	1	2	3	4	5	6	7
RST N	C7	CF	D7	DF	E7	EF	F7	FF

2.5. Команды управления стеком, вводом – выводом и микропроцессором

Команды этой группы состояние флагов не изменяют за исключением команды ***POP PSW*** восстанавливающей все флаги.

Операция	Мнемоника	Кодкоманды	Действие
Добавление в стек	PUSH rr		((SP)-1) ← (r r_H), ((SP)-2) ← (r r_L), (SP) ← (SP)-2
Удаление из стека	POP rr		(r r_L) ← ((SP)), (r r_H) ← ((SP)+1), (SP) ← (SP)+2
Обмен	XTHL	E3	(L) → ((SP)), (H) → ((SP)+1)

Передача	SPHL	F9	(SP) ← (HL)
Ввод	IN ?	DB	(A) ← (порт ?)
Вывод	OUT ?	D 3	(порт ?) ← (A)
Разрешение	EI	FB	Прерывания разрешены
Запрещение	DI	F3	Прерывания запрещены
Останов	HLT	76	Процесс остановлен
Нет операции	NOP	00	(PC) ← (PC)+1

2. КЛАВИВТУРА УОУ. ОСНОВНЫЕ РЕЖИМЫ РАБОТЫ С УОУ

3.1. Назначение клавиш

Управление работой УОУ осуществляется с помощью клавиатуры из 25 клавиш.

Назначение 9 командных клавиш:

RST формирует команду сброса МП RESET;

MEM включает режим записи данных в ячейку ОЗУ;

ADDR включает режим задания адреса ячейки памяти;

CLR восстанавливает первоначальное значение адреса или данных,

если после их ввода не нажимались другие командныеклавиши;

NEXT увеличивает на единицу адрес индицируемой ячейки памяти

или регистра

REG отображает содержимое 8-битного регистра МП;

STEP запускает в шаговом режиме очередную команду программы;

BRK задает адрес контрольной точки в программе;

RUN запускает программу на выполнение. При этом останов

произойдет на введенной контрольной точке или соответствующей команде.

16 клавиш данных служат для ввода 16-ричных цифр от 0 до F. Эти же клавиши задают имена регистров и регистровых пар: A, B, C, D, E, 8/H, 9/L, F обозначают соответственно аккумулятор, регистры общего назначения и регистр флагов.

1/P обозначает указатель стека *SP*,

2/T обозначает вершину стека *SI*.

Ячейки памяти с 0000_{16} до $03FF_{16}$ отведены под постоянное запоминающее устройство (ПЗУ). В нем хранится программа-монитор, запускаемая в работу сигналами от клавиатуры. Служебная область монитора расположена в оперативном запоминающем устройстве (ОЗУ). Ее адресное поле с $83E1_{16}$ до $83FF_{16}$. Таким образом, в распоряжении пользователя остается не все ОЗУ с адресами 8000_{16} + $87FF_{16}$, а два диапазона ячеек ОЗУ: 8000_{16} + $83E0_{16}$ и 8400_{16} + $87FF_{16}$.*

Клавиатура и монитор позволяют реализовать различные режимы работы:

ввод программы пользователя в ОЗУ;

выполнение программы в шаговом режиме, в режиме с остановом по контрольным точкам и в непрерывном режиме;

запись программы на магнитную ленту магнитофона;

считывание программы с магнитной ленты.

* Рекомендуемое размещение программы – с адреса 8200.

3.2. Чтение и запись содержимого программы и регистров.

Пусть *v*, *v* – 16-ричные цифры 0 + *F*;

N – любая из 16 клавиш данных.

1. Содержимое ячейки с адресом 8200 выводится на дисплей клавишей *RST* в виде

1	2	3	4	5	6	7	8
8	2	0	0			α	β

Содержимое ячейки с адресом *NNNN* выводится на дисплей клавишами *ADDR NNNN* в виде

1	2	3	4	5	6	7	8
N	N	N	N			α	β

Последующее нажатие клавиши *NEXT* индицирует адрес и содержимое следующей ячейки памяти.

Повторное нажатие клавиши *MEM* выводит на дисплей информацию из предыдущей ячейки.

2. Перед записью программы нужной ячейки ОЗУ устанавливается клавишами *ADDR NNNN MEM* в виде

1	2	3	4	5	6	7	8
N	N	N	N			α	β

(в 6-м разряде дисплея высвечивается точка). После этого данные вводятся нажатием одной или двух клавиш

данных; в 7-м и 8-м разрядах дисплея появляются новые данные.

Переход к следующей ячейке ОЗУ осуществляется клавишей *NEXT.*

Повторные нажатия клавиши *MEM* уменьшает адрес ячейки на единицу.

3. Пусть X-любой из регистров A, B, C, D, E, F, H, L. Содержимое регистра X выдается на дисплей клавишами

REG X в виде

1	2	3	4	5	6	7	8
N	N	N	N	X	-	α	β

(при этом в 1-4 разрядах дисплея сохраняется содержимое программного счетчика NNNN).

Нажатие клавиши NEXT выводит циклически на дисплей содержимое очередного регистра A,B,C,D,E,F,H,L,A,B,C,…

Для ввода данных в регистр X следует нажать одну или две клавиши данных.

4. Имя пары регистров отображаются в 5-6 разрядах дисплея, а их содержимоев1-4 разрядах дисплея с помощью клавиш ADDR R MEM, где

Значения R	B	D	8/H	1/P	2/T
Пара регистров	BC	DE	HL	SP	SF

3.3. Ввод контролъных точек.

Ввод контролъной точки по адресу NNNN с числом ее безостановочных проходов NN($NN<FF_{16}$) осуществляется нажатием клавиш ADDR NNNN BRR NN и отображается на дисплее в виде

1	2	3	4	5	6	7	8
N	N	N	N	B	P	N	N

(следует помнитъ, что нажатие клавиши RST аннулирует все контрольные точки).

3.4. Индикация ошибок

Запись данных в ОЗУ разрешена, если в 6-м разряде дисплея (нажатием клавиши MEM) высвечена точка. Код ошибки

1	2	3	4	5	6	7	8
E	r	r					

индицируется при попытках:

1) записать байт в ПЗУ или в несуществующую ячейку ОЗУ;
2) установить несуществующее имя регистра;
3) ввести цифру, если перед этим не была нажата из клавиш ADDR, MEM, REG, BRK;
4) запустить программу при адресе, меньшем 4 цифр, и т.д.

Если на дисплее появился код ошибки, то можно восстановить значение:

1) счетчика команд и саму команду клавишей CLR или ADDR;

2) ячейку памяти и ее адрес клавишей MEM.

3.5. Режимы выполнения программы.

Перед выполнением программы в любом режиме следует клавишами ADDR, VNNN задать ее начальный адрес.

Если тумблер выбора режима установлен в позицию ОТЛАДКА, то прерывание и вызов монитора происходит после выполнения каждой команды программы пользователя.

3.5.1. В шаговом режиме останов происходит после выполнения каждой команды. Перед пошаговым выполнением программы тумблер выбора режима следует установить в позицию ОТЛАДКА. Очередная команда программы выполняется по нажатию клавиши STEP. После выполнения этой команды новое значение программного счетчика PC отображается в 1-4 разрядах дисплея, а соответствующее PC содержимое ячейки памяти отображается в 7-8 разрядах дисплея.

После выполнения последней команды следует просмотреть регистры и ячейки ОЗУ с результатами.

В шаговом режиме можно проследить изменения содержимого любого регистра X=A,B,C,D,E,F,H,L.

Для этого перед выполнением первой команды следует нажать клавиши

REG X.

3.5.2. Монитор УОУ позволяет перед пуском программы клавишей RUN создать режим останова по контрольным точкам, т.е. определить адреса, где выполнение программы необходимо прервать для проверки промежуточных результатов. Контрольные точки, каждая из которых занимает 4 байта, записываются в ту же область ОЗУ, где размещается стек.

Напомним, что ввод по адресу NNNN контрольной точки с числом ее безостановочных проходов NN(NN=<FF_{16})

Если введены контрольные точки, то при исполнении программы в режиме RUN проверяется, соответствует ли содержание программного

счетчика какой-либо контрольной точке. При их совпадении монитор уменьшает на единицу содержимое числа проходов данной контрольной точки и, если оно равно нулю, то происходит останов программы пользователя.

Программа останавливается до выполнения команды, адресуемой контрольной точкой; при этом вызываются входящие в состав монитора подпрограммы клавиатуры и дисплея.

Для выполнения программы с остановами по контрольным точкам необходимо:

1) установить тумблер выбора режима ОТЛАДКА;

2) ввести контрольные точки;

3) клавишами ADDR, NNNN задать начальный адрес программы;

4) клавишей RUN пустить программу на выполнение.

В этом режиме программа после выполнения каждой команды автоматически прерывается монитором для контроля точек останова.

Клавиатура и дисплей блокируются до момента, пока текущий адрес программы не сравняется с адресом контрольной точки и число проходов данной контрольной точки не станет равным нулю. После этого будет вызван монитор, а на дисплее отобразится адрес той контрольной точки, на которой произошел останов:

1	2	3	4	5	6	7	8
N	N	N	N				

Нажатие клавиши BRK выводит на дисплей информацию об этой контрольной точке:

1	2	3	4	5	6	7	8
N	N	N	N	B	R	O	O

1	2	3	3	5	6	7
N	N	N	N	B	P	N

При останове в этом режиме, последовательно нажимая клавишу NEXT, можно циклически просмотреть все контрольные точки с переходом от последней к первой. Появившуюся на дисплее информацию о контрольной точке можно изменить, установив новое число проходов NN.

или исключить эту контрольную точку клавишей CLR; при нажатии этой клавиши индицируется адрес следующей контрольной точки. Если контрольных точек больше нет, то дисплей принимает вид

1	2	3	4	5	6	7	8
				B	P		

Примечания.

1. При вводе или исключении контрольных точек стек сдвигается. Поэтому, если программа пользователя восстанавливает указатель стека, нельзя добавлять или исключать контрольные точки.

2. Необходимо учитывать, что нажатие клавиши RST исключает все контрольные точки.

Пример. В ОЗУ введена следующая программа:

8200	AF		XRA A
1	3C	M:	INR A
2	3C		INR A
3	3C		INR A
4	C3		JMP M
5	01		JMP M
6	82		JMP M

После ввода программы нажаты клавиши:

ADDR	8204	BRK	00	ADDR	8200	RUN
ADDR	8204	BRK	01	ADDR	8200	RUN
ADDR	8204	BRK	02	ADDR	8200	RUN
ADDR	8204	BRK	03	ADDR	8200	RUN

Вид дисплея

1	2	3	4	5	6	7	8
8	2	0	4	A	-	0	3
8	2	0	4	A	-	0	6
8	2	0	4	A	-	0	9
8	2	0	4	A	-	0	C

В данном примере после останова адрес контрольной точки 8204 отображается в 1-4 разрядах дисплея; в 5-м разряде указывается наименование регистра A, а в 7-8 разрядах отображается содержимое этого регистра после последнего выполнения команды 8203.

Кроме описанного, имеется возможность с помощью клавиши BRK прекратить работу программы после выполнения заранее заданного числа команд.

Для этого необходимо после ввода программы в ОЗУ:

1) клавишами ADDR 83E6 BRK NN записать в ячейку с адресом 83E6 уменьшенное на единицу число подлежащих выполнению команд NN

1	2	3	4	5	6	7	8
8	3	E	6	B	P	N	N

2) клавишами ADDR NNNN установить начальный адрес программы;

3) клавишей RUN пустить программу на выполнение.

Пример. В ОЗУ введена следующая программа:

8200	AF	XRA
1	3C	INR A
2	3C	INR A

3	3C	INR A
4	3C	INR A

Вид дисплея

1	2	3	4	5	6	7	8
8	2	0	1	A	-	0	0
8	2	0	2	A	-	0	1
8	2	0	3	A	-	0	2
8	2	0	4	A	-	0	3

После ввода программы нажать клавиши

ADDR	83E6	BRK	00	ADDR	8200	RUN
ADDR	83E6	BRK	01	ADDR	8200	RUN
ADDR	83E6	BRK	02	ADDR	8200	RUN
ADDR	83E6	BRK	03	ADDR	8200	RUN

В данном примере в 1-4 разрядах дисплея отображается адрес команды, перед выполнением которой произошел останов; в 5-м разряде указывается наименование регистра А, а отображаемое в 7-8 разрядах содержимое этого регистра соответствует значению NN. 25

3.5.3. При выполнении программы в непривычном режиме УОУ работает без подключения монитора, что ускоряет вычисления. Отключение монитора производится установкой тумблера выбора режима в позицию ПРОГОН.

Для того чтобы по окончании программы произошли прерывание и вызов монитора, включающего индикацию, необходимо в качестве команды останова использовать не HLT, а команду RST4(ее код

$E7_{16}$).Иначе, например, при останове по команде HLT(76_{16}), монитор не будет вызван и вывод информации на дисплей окажется невозможным.

Для выполнения программы в непрерывном режиме необходимо:

1) установить тумблер выбора режима в позицию ПРОГОН;

2) клавишами DDDR NNNN задать начальный адрес программы;

3) пустить программу клавишей RUN

Сводка режимов выполнения программы

Режим выполнения программы	Положение тумблера выбора режима	Клавиша пуска программы	16-ричной код команды останова
1. Шаговый	ОТЛАДКА	STEP	76_{16}
2.С остановом по контрольным точкам	ОТЛАДКА	RUN	76_{16}
3.Непрерывный	ПРОГОН	RUN	$E7_{16}$

3.6. Рекомендации по использованию прерываний

Каждое прерывание имеет свой вектор, по которому МП определяет начальный адрес программы обработки прерывания.

Как отмечалось, по команде RST N(0=<N=<7) МП переходит на строго фиксированные адреса вида 8∗N, лежащие в диапазоне 0000_{16}-0038_{16}. Но в УОУ эта область памяти занята монитором, в связи с чем начальные адреса программ обработки прерываний хранятся в специальных ячейках-диспетчерах, куда управление передается из ячеек с адресами вида 8∗N; содержимое диспетчера при поступлении кода прерываний загружается в программный счетчик, а также может быть изменено пользователем.

Монитор использует 3 вектора прерывания:

RST 0 – для начального запуска программ;

RST 4 – для вызова монитора пользователем;

RST 7 – для вызова монитора по прерыванию команды пользователя.

Остальные 5 векторов прерывания пользователь может задействовать без ущерба для работы монитора. Следовательно, для использования прерываний пользователь должен предварительно загрузить соответствующими адресами ячейки – диспетчеры:

(83F4,83F5) – для RST 1;

(83F2,83F3) – для RST 2;

(83F0, 83F1) – для RST 3;

(83EC, 83ED) – для RST 5;

(83EA,83EB) – для RST 6

(в ячейку – диспетчер с меньшим адресом загружается младший байт адреса перехода).

3. ЗАДАЧИ, РЕКОМЕНДУЕМЫЕ ДЛЯ САМОСТОЯТЕЛЬНОГО ВЫПОЛНЕНИЯ И ВЫРАБОТКИ НАВЫКОВ РАБОТЫ С УОУ

4.1. Использование регистров и ячеек памяти

Программа 4.1(выполняется в шаговом режиме)

Адрес	Код	Метка	Мнемокод	Комментарий
8200	3E		MVI A	
8201	19		19	Переслать байт 19 в регистр A
8202	06		MVI B	
8203	25		25	Переслать байт 25 в регистр B
8204	80		ADD B	Прибавить к A регистр B
8205	21		LXI HL	
8206	51		51	Загрузить двойной регистр HL
8207	82		82	Адресом ячейки 8251
8208	70		MOV M,B	Переслать регистр B в ячейку 8251
8209	3D		DCR A	Уменьшить на 1 регистр A
820A	16		MVI D	
820B	47		47	Переслать байт 47 в регистр D

820C	82		ADD D	Прибавить к А регистр D
820D	4F		MOV C,A	Переслать регистр А в регистр С
820E	2C		INR L	Увеличить на 1 регистр L
820F	71		MOV M,C	Переслать регистр С в ячейку 8252
8210	81		ADD C	Прибавить к А регистр С
8211	5F		MOVE,A	Переслать регистр А в регистр Е

Изменение содержимого регистра А после команд:

8200	8204	8209	820C	8210
19_{16}	$3E_{16}$	$3D_{16}$	84_{16}	08_{16}

Содержимое регистров и ячеек после выполнения программ:

A	B	C	D	E	H	L	8251	8252
08_{16}	25_{16}	84_{16}	47_{16}	08_{16}	82_{16}	52_{16}	25_{16}	84_{16}

4.2 Сложение двух двухбайтных чисел

Программа 4.2(выполняется в шаговом режиме)

Адрес	Код	Метка	Мнемокод	Комментарий
8200 8201	06 44		MVI B 44	Переслать байт 44 в регистр В
8202 8203	0E 33		MVI C 33	Переслать байт 33 в регистр С
8204 8205	16 47		MVI D 47	Переслать байт 47 в регистр D
8206 8207	1E 53		MVI E 53	Переслать байт 53 в регистр Е
8208	EB		XCHG	Обменять пару регистров DE с парой HL

8209	09		DAD BC	Прибавить к паре регистров HL содержимое BC

Команда сложения регистровых пар DAD проводим суммирование только с двойным регистром HL. Поэтому перед сложением двухбайтных чисел,

Хранящихся в двойных регистрах BC и DE, предварительно осуществляется обмен регистров DE и HL.

Результат сложения помещается в паре регистров HL

4433_{16}

+

4753_{16}

$8B86_{16}$

Используя непосредственную загрузку пары регистров, можно следующим образом сократить программу:

Адрес	Код	Метка	Мнемокод	Комментарий
8200	01		LXI BC	Загрузить двойной регистр
8201	33		33	BC двухбайтовым числом
8202	44		44	4433
8203	21		LXI HL	Загрузить двойной регистр
8204	52		53	HL двухбайтовым числом
8205	47		47	4753
8206	09		DAD BC	Прибавить к паре HL содержимое BC

4.3 Сложение массива однобайтных чисел

Пусть необходимо найти сумму чисел ***a1***,***a2***,……***a16***, хранящихся в ячейках 8301,8302,…..,8310. предполагается, что сумма чисел не превосходит 255 ,а засылка этих чисел предварительно произведена с пульта клавишами:

ADDR 8301 MEM ***a1***NEXT ***a2***NEXT…..***a16***NEXT

Блок-схема программы сложения массива однобайтных чисел показана на рис.3

Программа 4.3(выполняется в непрерывном режиме)

8200 8201 8202	21 01 83		LXI HL 01 83	Загрузить двойной регистр HL адресом числа ***a1*** ,равным 8301
8203 8204	06 10		MVI B 10	Переслать число слагаемых 10_{16} в регистр B
8205	AF	SUM:	XRA A	Очистка регистра A
8206	86		ADD M	Прибавить к A содержимое ячейки M
8207	23		INX HL	Увеличить на 1 двойной регистр HL
8208	05		DCR B	Уменьшить на 1 регистр B
8209 820A 820B	C2 06 82		JNZ 06 82	Если флаг нуля не установлен, то Перейти на метку SUM
820C	E7		RST 4	Рестарт 4

4.4 Сложение массива однобайтных чисел, если их сумма больше 255

Пусть необходимо найти сумму чисел ***a1,a2,……a16,***предварительно записанных в ячейках 8300,8301,830P.Предполагается, что сумма чисел превосходит 255, в связи с чем старший байт суммы накапливается в регистре C ,а младший – в регистре A

Блок-схема программы сложения массива однобайтных чисел ,если их сумма больше 255 ,представлена на рис. 4.

Программа 4.4 (выполняются в непрерывном режиме)

Адресс	Код	Метка	Мнемокод	Комментарий
8200 8201 8202	21 00 83		LXI HL 00 83	Загрузить двойной регистр HL адресом числа ***a1*** ,равным 3300
8203 8204	06 10		MVI B 10	Переслать число слагаемых 10_{16} в регистр B
8205	AF		XRA A	Очистка регистра A
8206	4F		MOV C,A	Очистка регистра C
8207	86	SUM	ADD M	Прибавить к A содержимое ячейки M
8208 8209 820A	D2 0C 82		JNC 0C 82	Если флаг переноса не установлен, то перейти на метку DLT
820B	OC		INR C	Увеличить на I регистр C
820C	23	DLT	INX HL	Увеличить на 1 двойной регистр HL
820D	05		DCR B	Уменьшить на 1 регистр B
82OE 820F 8210	C2 07 82		JNZ 07 82	Если флаг нуля не установлен, то Перейти на метку SUM
8211	E7		RST 4	Рестарт 4

4.5.Умножение однобайтных чисел.

При вычислении произведения необходимо осуществить многократное суммирование со сдвигом вправо суммы при одновременной проверке содержимого разрядов множителя, начиная со стороны его младшего разряда. При этом, если в очередном разряде множителя записан 1, то множимое прибавляется к сумме и сумма сдвигается вправо на один разряд, а если в разряде записан 0, то производится только сдвиг суммы.

Блок-схема программы умножения однобайтных чисел приведена на рис. 5.

Программа 4.5 (выполняется в непрерывном режиме)

Адрес	Код	Метка	Мнемокод	Комментарии
8200 8201	16 *		MVI D *	Загрузить байт множимого в регистр D
8202 8203	1E *		MVI E *	Загрузить байт множителя в регистр E
8204 8205 8206	01 00 00		LXI BC 00 00	Очистить регистры будущего произведения BC
8207 8208	3E 01		MVI A 01	Загрузить регистр A маску $0000\ 00001_2$
8209 820A 820B * 820C	A7 F5 A3 78	M1:	ANA A PUSH PSW ANA E MOV A, B	Очистить флаг переноса CY Отослать в стек маску Выделить очередной разряд множителя Вызвать в A старший байт результата
820D 820E 820F	CA 11 82		JZ 11 82	Если флаг нуля установлен, то перейти на метку M2
8210	82	M2:	ADD D	Прибавить можемое к результату

8211	1F		RAR	Сдвиг вправо; младший бит -> CY
8212	47		MOV B, A	Сохранить старший байт в регистре B
8213	79		MOV A, C	Вызвать в A младший байт результата
8214	1F		RAR	Сдвиг вправо; CY -> в старший бит
8215	4F		MOV C, A	Соранить младший байт в регистре C
8216	F1		POP PWS	Вызвать из стека маску
8217	17		RAL	Сдвинуть текущую маску влево
8218	D2		JNC	Если флаг переноса не установлен, то перейти на метку M1
8219	0A		0A	
821A	82		82	
821B	E7		RST 4	Рестарт 4

4.6. «Движущийся» текст

Дисплей из 8 индикаторов при включенном УОУ отражает содержание ячеек с адресами 83F8-83FF.Индикатор (см. рис.6)состоит из 8 светодиодов, каждый из которых светится, если соответствующий его номеру бит ячейки равен 1. Например, для отображения цифры 9 соответствующий байт должен иметь вид:

Номера битов	8 7 6 5	4 3 2 1
Содержание битов	0 1 1 0	1 1 1 1
16-ричная запись байта	6	F

Таким образом, для перемещения на дисплее, например, текста:

«СССР – 1988 г.»

следует, во-первых; предварительно разместить в каких-либо ячейках (например, 8301-830C) следующие:

коды символов: 39 39 39 73 40 06 6F 7F 7F D0 00 00

символы: С С С Р - 1 9 8 8 г.

и, во-вторых, организовать в ячейках 83F8-83FF циркуляцию указанных кодов по кольцу справа налево.

Блок-схема программы «движущегося» текста показана на рис.7.

Программа 4.6 (выполняется в непрерывном режиме)

Адрес	Код	Метка	Мнемокод	Комментарий
8200 8201	OE OB	M1	MVI C OB	Заслать счетчик длины текста(OB) в регистр C
8202 8203 8204	11 01 83		LXI DE 01 83	Загрузить двойной регистр DE адресом 8301
8205 8206	62 6B		MOV H, D MOV L, E	Переслать адрес 8301 в двойной регистр HL
8207	7E	M2:	MOV A,M	Вызвать код символа в регистр A
8208 8209 820A	32 FF 83		STA FF 83	Переслать код символа в ячейку с адресом 83FF
820B 820C 820D	CD 16 82		CALL 16 82	Обратиться к подпрограмме TIME
820E	0D		DCR C	Уменьшить на 1 регистр C
820F 8210 8211	C2 07 82		JNZ 07 82	Если флаг нуля не установлен, то перейти на метку M2
8212 8213	C3 00		JMP 00	Переход на метку M1

8214	82		82	
8215	E7		RST 4	Рестарт
8216 8217	06 AO	TIME	MVI B A0	Заслать счетчик внешнего цикла AO в регистр B
8218 8219	3E FF	M3:	MVI A FF	Заслать счетчик внутреннего цикла FF в регистр A
821A	3D	M4:	DCR A	Уменьшить на 1 регистр A
821B 821C 821D	C2 1A 82		JNZ 1A 82	Если флаг нуля не установлен, то перейти на метку M4
821E	05		DCRB	Уменьшить на 1 регистр B.
821F 8220 8221	C2 18 82		JNZ 18 82	Если флаг нуля не установлен, то перейти на метку M3
8222 8223	06 07		MVI B 07	Заслать счетчик числа перемещаемых кодов в регистр B
8224 8225 8226	21 F9 83		LXI HL F9 83	Загрузить двойной регистр HL адресом 2-го кода символа 83F9
8227	7E	M5:	MOV A, M	Вызвать код символа в регистр A
8228	2B		DCX HL	Уменьшить на 1 двойной регистр HL
8229	77		MOV M,A	Переслать код символа на место предыдущего
822A	23		INX HL	Восстановить адрес вызванного кода символа
822B	23		INX HL	Подготовить адрес следующего кода символа
822C	05		DCR B	Уменьшить на 1 регистр B
822D 822E 822F	C2 27 82		JNZ 27 82	Если флаг нуля не установлен, то перейти на метку M5
8230	13		INX DE	Увеличить на 1 двойной регистр DE

8231
 	
8232
	62
	
	6B
			MOV H, D
			
			MOV L, E
				Переслать адрес кода символа
				
				в двойной регистр HL

8233	C9		RET	Выход из подпрограммы

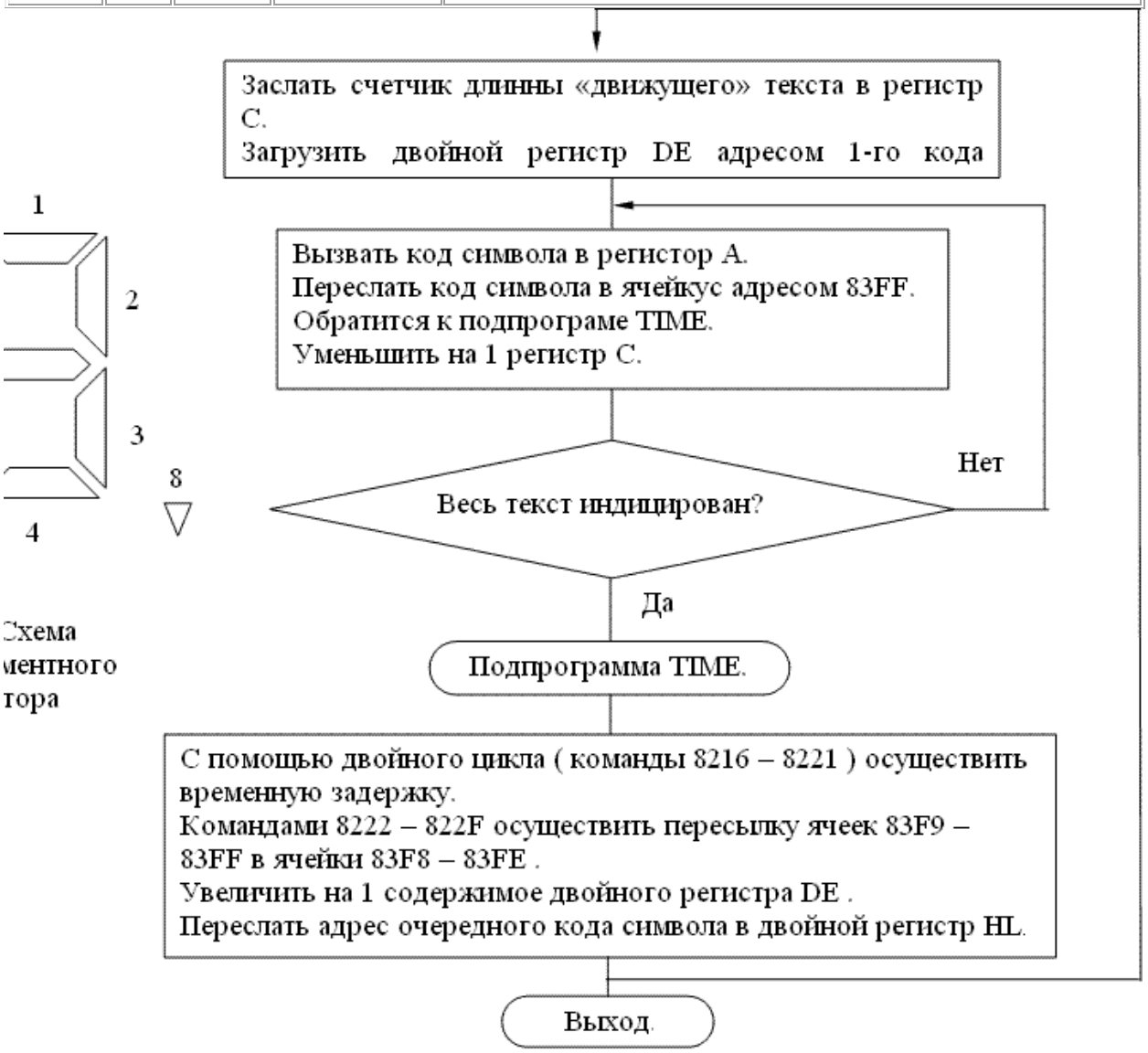


Рис.7. Блок – схема программы «движущегося» текста

Приложение Алфавитный перечень мнемокодов МП К580.

ACI	v	Прибавить к аккумулятору значение V и флаг переноса (ADDITION CARR IMMEDIATE)
ADC	r	Прибавить к аккумулятору флаг переноса и содержимое регистра r или ячейки памяти M (ADDITION CARRY)

ADD	r	Прибавить к аккумулятору содержимое регистра r или ячейки памяти M (ADDITION)
ADI	v	Прибавить к аккумулятору значение V (ADDITION IMMEDIATE)
ANA	r	Логическое умножение «И» аккумулятора и содержимого регистра r или ячейки памяти M (AND)
ANI	v	Логическое умножение «И» аккумулятора и значения V (AND IMMEDIATE)
CALL	aa	Вызвать подпрограмму
		Вызвать подпрограмму, если:
CZ	aa	Флаг нуля установлен(CALL IF ZERO)
CNZ	aa	Флаг нуля не установлен(CALL IF NOT ZERO)
CP	aa	Аккумулятор положительный(CALL IF PLUS)
CM	aa	Аккумулятор отрицательный(CALL IF MINUS)
CC	aa	Флаг переноса установлен(CALL IF CARRY)
CNC	aa	Флаг переноса не установлен (CALL IF NOCARRY)
CPE	aa	Флаг паритета установлен(CALL IF PARITY EVEN)
CPO	aa	Флаг паритета не установлен(CALL IF PARITY ODD)
CMA		Дополнить аккумулятор(COMPLEMENT ACCUMULATOR)
CMC		Дополнить флаг переноса(COMPLEMENT CARRY)
CMP	r	Сравнить аккумулятор с содержимым регистра r или ячейки памяти M(COMPARE)
CPI	v	Сравнить аккумулятор со значением V(COMPARE IMMEDIATE)
DAA		Десятичная коррекция аккумулятора(DECIMAL ADJUST ACCUMULATOR)
DAD	r r	Прибавить к паре регистров HL содержимое пары регистров rr или указателя стека (DOUBLE ACCEPTATION DATA)
DCR	r	Уменьшить регистр r или ячейку памяти M (DECREMENT REGISTER)
DCX	r r	Уменьшить пару регистров rr или указатель стека (DECREMENT EXTENDED)
DI		Запретить прерывания(DISABLE INTERRUPT)
EI		Разрешить прерывания(ENABLE INTERRUPT)
HLT		Остановить процессор HALT
IN	v	Ввести в аккумулятор байт из порта с адресом V(INPUT)

INR	r	Увеличить регистр r или ячейку памяти M (INCREMENT REGISTER)
INX	r r	Увеличить пару регистров rr или указатель стека (INCREMENT EXTENDED)
JMP	aa	Перейти по адресу aa (JUMP)

Перейти по адресу, если:

JZ	aa	Флаг нуля установлен(JUMP IF ZERO)
JNZ	aa	Флаг нуля не установлен(JUMP IF NOT ZERO)
JP	aa	Аккумулятор положительный(JUMP IF PLUS)
JM	aa	Аккумулятор отрицательный(JUMP IF MINUS)
JC	aa	Флаг переноса установлен(JUMP IF CARRY)
JNC	aa	Флаг переноса не установлен(JUMP IF NO CARRY)
JPE	aa	Флаг паритета установлен (JUMP IF PARITY EVEN)
JPO	aa	Флаг паритета не установлен (JUMP IF PARITY ODD)

LDA	aa	Загрузить в аккумулятор байт, заданный адресом aa (LOAD ACCUMULATOR)
LDAX	r r	Загрузить в аккумулятор байт, заданный адресом, содержащимся в паре регистров BC и DE (LOAD ACCUMULATOR EXTENDED)
LHLD	aa	Загрузить в пару регистров HL байты, заданные адресами aa и aa+1 (LOAD HL DIRECT)
LHI	vv	Загрузить пару регистров 16-битным значением vv (LOAD EXTENDED IMMEDIATE)
MOV	r1,r2	Переслать содержимое (MOVE DATA) регистра
MOV	m,r	R2 в регистр r2 (ячейки M в регистр r)
MOV	r,m	(регистра r в ячейку M)
MVI	r,v	Переслать значение v в регистр r или ячейку памяти M (MOVE IMMEDIATE DATA)
NOP		Нет операции (NO OPERATION)
ORA	r	Логическое сложение «ИЛИ» аккумулятора с содержимым регистра r или ячейки памяти M(OR)
ORI	v	Логическое сложение «ИЛИ» аккумулятора со значением v (OR IMMEDIATE)

OUT	v	Послать содержимое аккумулятора в порт с адресом v (OUT PUT)
PCHL		Переход по адресу, указанному в паре регистров HL (PROGRAM COUNTER)
POP	r r	Прочитать из стека в заданную пару регистров (PULL UA)
PUSH	r r	Записать заданную пару регистров в стек (PUSH DOWN)
RAL		Арифметический сдвиг аккумуляторов влево (ROTATE ARITHMETIC LEFT)
RAR		Арифметический сдвиг аккумуляторов вправо (ROTATE ARITHMETIC RIGHT)
RET		Возвратиться из подпрограммы (RETURN)
		Возвратиться из подпрограммы, если
RZ		Флаг нуля установлен (REURN IF ZERO)
RNZ		Флаг нуля не установлен(RETURN IF NO ZERO)
RP		Аккумулятор положительный
RM		Аккумулятор отрицательный
RC		Флаг переноса установлен
RNC		Флаг переноса не установлен
RPE		Флаг паритета установлен
RBO		Флаг паритета не установлен
RLC		Циклически сдвинуть аккумулятор влево и во флаг переноса
RRC		Циклически сдвинуть аккумулятор вправо и во флаг переноса
RST	n	Повторный пуск, вызвать подпрограмму по адресу 8N
SBB	r	Вычесть из аккумулятора регистр r или ячейку памяти с учётом переноса (SUBTRACT BORROW)
SBI	v	Вычесть из аккумулятора значение v с учётом переноса (SUBTRACT BORROW IMMEDIATE)
SHLD	aa	Записать содержимое пары регистров HL в ячейки памяти с адресами aa и aa+1(STORE HL DIRECT)
SPHL		Переслать содержимое пары регистров HL в стек (STACK POINTER)
STA	aa	Загрузить содержимое аккумулятора в ячейку памяти с адресом aa (STORE ACCUMULATOR)
STAX	aa	Загрузить содержимое аккумулятора в ячейку с адресом, указанным в паре регистров BC или DE (STORE

		ACCUMULATOR EXTENDED)
STC		Установить флаг переноса (SET CARRY)
SUB	r	Вычесть из аккумулятора регистр r или ячейку памяти M(SUBTRACT)
SUI	v	Вычесть значение v из аккумулятора(SUBTRACT IMMEDIATE)
XCHG		Обменять пару регистров DE с парой HL (EXCHANGE)
XRA	r	Выполнить логическую операцию «исключающее или» над аккумулятором и содержимым регистра r или ячейки памяти M (EXCLUSIVE OR)
XRI	v	Выполнить логическую операцию «исключающее или» над аккумулятором и значением v (EXCLUSIVE OR IMMEDIATE)
XTHL		Обменять пару регистров HL с содержимым вершины стека (EXCHANGE HL)

Задания для лабораторной работы «Адресация (работа с памятью), команды пересылки, сложения, вычитания»

1. Число 2A загрузить в регистр B. Вычитать из него последовательно числа 03, 05, 07, 09, 0B, $0D_{16}$. Промежуточные результаты сохранять в ячейках с адресами 8300, … , 8305_{16}:

2A-03 -> 8300
(2A-03)-05->8301…

…

2. Число 19_{16} загрузить в регистр C. Прибавлять к нему последовательно числа 21, 23, 25, 27, 29, $2B_{16}$. Промежуточные результаты сохранять в ячейках 8310, 8313, 8316, …, 8325_{16}:

19+21-> 8310
(19+21)+23-> 8313…

…

3. Число $B5_{16}$ загрузить в ячейку памяти 8300. Прибавлять к нему последовательно числа 05, 07, 09, 0B, $0C_{16}$. Промежуточные результаты сохранять в ячейках 8301, 82FF,…, $82F9_{16}$:

B5+05->8301
(B5+05)+07->82FF…

…

4. Число $3F_{16}$ загрузить в ячейку памяти 8330. Прибавлять к нему поочередно числа BA, BB, …,$C2_{16}$. Промежуточные результаты сохранять в ячейках 8328, 8326, 8324, …:

3F+BA-> 8328
3F+BB-> 8326…

…

5. Число FF_{16} загрузить в регистр D. Вычитать из него поочередно числа BF, BE, BD, … , $B8_{16}$. Промежуточные результаты сохранять в ячейках 8340, 8342, …, $834E_{16}$:

FF-BF -> 8340

FF-BE ->8342…

…

6. Число $A5_{16}$ загрузить в ячейку памяти 8330. Вычитать из него последовательно числа 05, 0A, 0F, 14, 19, 1E, 23_{16}. Промежуточные результаты сохранять в ячейках 8332, 8334,…:

A5-05 -> 8332,

(A5-05) –0A ->833…

…

7. Число $B5_{16}$ загрузить в ячейку памяти 8350. Вычитать из него поочередно числа 02, 05, 08, 0B, 0E, 11_{16}. Промежуточные результаты сохранять в ячейках 834E, 834C,…, 8344_{16}:

B5-02 -> 834E

B5-05 -> 834C…

…

8. Число AD_{16} загрузить в регистр B. К числу AD_{16} прибавлять последовательно на 01, 03, …, 09_{16}. Промежуточные результаты сохранять в ячейках 8300, 8302,…:

AD+01 -> 8300

(AD+01)+03 -> 8302…

…

9. Число $F5_{16}$ загрузить в ячейку памяти 8300. Вычитать из него последовательно числа 06, 05, 04, 03, 02_{16}. Промежуточные результаты сохранять в ячейках 8327, 8325,…:

F5-06 ->8327

(F5-06)-05 ->8325, …

…

10. Число FA_{16} загрузить в регистр L. Вычитать из него последовательно числа $1B_{16}$, 1A, 19, 18, 17_{16}. Промежуточные результаты сохранять в ячейках 8300, 8302…:

FA – 1B ->8300

(FA – 1B)-1A ->8302…

…

11. Число $6B_{16}$ загрузить в регистр C. Из числа $6B_{16}$ вычитать числа 05, 07, …,$0D_{16}$. Промежуточные результаты сохранять в в ячейках 8301, 8303,…:

6B-05 ->8301

(6B-05)-07 ->8303…

…

12. Число 2A загрузить в регистр B. Прибавлять к нему поочередно числа 03, 05, 07, 09, 0B, $0D_{16}$. Промежуточные результаты сохранять в ячейках с адресами 8300, … ,8305_{16}:

2A+03 -> 8300

2A+05->8301…

…

13. Число CC_{16} загрузить в регистр C. Вычитать из него последовательно числа 05, 07, …,$0D_{16}$. Промежуточные результаты сохранять в в ячейках 8301, 8303,…:

CC-05 ->8301

(CC-05)-07 ->8303…

…

14 Число $F9_{16}$ загрузить в регистр C. Прибавлять к нему последовательно числа B1, B3, B5, B7, B9, BB_{16}. Промежуточные результаты сохранять в парах ячеек 8310- 8311, 8313-8314, 8316-8317, …, 8325-832616:

F9+B1-> ст.байт ->8310, мл. байт ->8311

(F9+B1)+B3-> ст.байт ->8313, мл. байт ->8314...

…

15 Число $3F_{16}$ загрузить в ячейку памяти 8330. Прибавлять к нему последовательно числа 01BA, 01BB, …,$01C2_{16}$. Промежуточные результаты сохранять в парах ячеек 8328-8329, 8326-8327, 8324-8325, …:

3F+01BA-> ст. байт ->8328, мл. байт ->8329

(3F+01BA)+01BB-> ст. байт ->8326, мл. байт ->8327...

…

16 Число FF_{16} загрузить в регистр D. Прибавлять к нему поочередно числа BF, BE, BD, … , $B8_{16}$. Промежуточные результаты сохранять в парах ячеек 8340-8341, 8342-8343, …, 834E-$834F_{16}$:

FF+BF -> ст. байт ->8340, мл. байт ->8341

FF+BE -> ст. байт ->8342, мл. байт ->8343...

…

17 Число $FF15_{16}$ загрузить в ячейки памяти 8330 (ст.байт) и 8331 – мл. байт. Вычитать из него последовательно числа 05, 0A, 0F, 14, 19, 1E, 23_{16}. Промежуточные результаты сохранять в парах ячеек 8332-8333, 8334-8335,…:

FF15-05 -> ст. байт ->8332, мл. байт ->8333

(FF15-05) –0A -> ст. байт ->8334, мл. байт ->8335...

…

18 Число $FFF5_{16}$ загрузить в ячейки памяти 8300 (ст.байт) и 8301 – мл. байт. Вычитать из него последовательно числа 06, 05, 04, 03, 02_{16}. Промежуточные результаты сохранять в парах ячеек 8327-8326, 8325-8324,…:

FFF5-06 -> ст. байт ->8327, мл. байт ->8326

(FFF5-06)-05 -> ст. байт ->8325, мл. байт ->8324...

…

19 Число $F51A_{16}$ загрузить в любые 2 регистра. Вычитать из него числа 1B, 1A, 19, 18, 17_{16}. Промежуточные результаты сохранять в ячейках 8300-8301, 8302-8303,…:

F51A – 1B -> ст. байт ->8300, мл. байт ->8301

(F51A – 1B)-1A -> ст. байт ->8302, мл. байт ->8303...

…

Федеральное агентство морского и речного транспорта
Федеральное государственное бюджетное образовательное
учреждение высшего образования
«Волжский государственный университет водного транспорта»

Кафедра информатики, систем управления
и телекоммуникаций

Т.И. Гаврилова

ОСНОВЫ ЦИФРОВОГО МОДЕЛИРОВАНИЯ

Учебно-методическое пособие
для студентов очного и заочного обучения
специальности 26.05.07-65
«Эксплуатация судового электрооборудования и средств автоматики»

Нижний Новгород
Издательство ФГБОУ ВО «ВГУВТ»
2016

УДК 004.942:621.3.011

Г

Гаврилова, Т.И.

Основы цифрового моделирования: учеб.-метод. пособие для студентов всех форм обуч. спец-ти 26.05.07-65 «Эксплуатация судового электрооборудования и средств автоматики» / Т.И. Гаврилова.–
Н. Новгород: Изд-во ФГБОУ ВО «ВГУВТ», 2016. – 47 с.

Даётся понятие модели и различных видов моделирования, в том числе математического. Демонстрируются приёмы моделирования на примере электрических цепей. Рассматриваются способы решения дифференциальных уравнений, являющихся широко используемым аппаратом построения математических моделей динамических систем. Приводятся примеры получения решения с помощью языка программирования Си, а также в среде математического пакета MathCAD 14.0 компании Parametric Technology Corporation.

Работа рекомендована к изданию кафедрой информатики, систем управления и телекоммуникаций (протокол №4 от 04.03.2016 г.).

© ФГБОУ ВО «ВГУВТ», 2016

Введение

Моделирование как вид деятельности, направленной на получение новых представлений об окружающем мире, используется давно. История развития человечества характеризуется постоянным совершенствованием этого метода познания.

Процесс моделирования требует выделения наиболее существенных для проводимого исследования свойств объекта. Так для изучения аэродинамических качеств модели самолета в аэродинамической трубе важно, чтобы модель имела геометрическое подобие оригинала.

Модель — это некий новый объект, который отражает существенные особенности изучаемого объекта, явления или процесса. Для одного объекта может быть построено множество моделей. Примерами моделей могут быть макеты, схемы, чертежи, словесные описания, математические формулы, карты, натурные модели судов, испытываемые в опытовом бассейне и т.д.

Существуют различные классификации моделей.

По способу представления все модели можно разбить на два больших класса: модели предметные (натурные) и модели информационные.

Предметные (натурные) модели воспроизводят геометрические, физические и другие свойства объектов в материальной форме (глобус, анатомические муляжи, модели кристаллических решеток, макеты зданий и сооружений, модели судов и др.).

Информационные модели представляют объекты и процессы в образной (мысленной, разговорной) или знаковой форме, выраженной средствами некоторого формального языка (графики, таблицы, математические формулы, компьютерные программы).

Классификация по фактору времени позволяет выделить модели статические и динамические.

Статическое моделирование служит для описания состояния объекта в фиксированный момент времени, а динамическое — для исследования объекта во времени.

В пособии рассматривается динамическое моделирование, т.е. моделирование объектов, функционирующих в непрерывном времени, непрерывно наблюдаемых и изменяющих свое состояние под воздействием внешних и внутренних причин.

1. Компьютерное моделирование

Развитие вычислительной техники привело к появлению нового метода научных исследований – *компьютерного цифрового моделирования,* который основывается на построении математических моделей для описания изучаемых объектов и использовании вычислительных машин для решения возникающих математических задач.

Моделирование на цифровых вычислительных машинах является одним из наиболее мощных средств исследования. Оно применяется при проектировании систем, в задачах прогнозирования, в автоматизированных системах управления, при решении вопросов оптимизации.

Компьютерное моделирование также позволяет изучать объекты, натурные эксперименты с которыми не целесообразны из-за соображений безопасности или дороговизны. Используя этот метод, можно узнать свойства моделируемого объекта, которые редко проявляются при обычных условиях функционирования, но могут проявиться при особых условиях, которые также могут быть смоделированы исследователем.

Этапы компьютерного моделирования:

1 этап. Получение математической модели.

Математическая модель – это модель объекта, процесса или явления, представляющая собой математические закономерности, с помощью которых описаны основные характеристики моделируемого объекта, процесса или явления.

Построение математической модели выполняется на основании законов, действующих в исследуемой системе, и завершается записью сформулированных в математических терминах представлений о связях между различными характеристиками системы.

2 этап. Выбор математического аппарата для анализа математической модели и программных средств для получения численных результатов решения задачи.

Математическое моделирование объектов различной природы часто приводит к получению одинаковых математических задач.

Такие типичные математические задачи рассматриваются исследователями как самостоятельный объект изучения.

3 этап. Проведение серии вычислительных экспериментов с целью выяснить, удовлетворяет ли модель критерию практики.

На этом этапе выясняется, согласуются ли расчетные значения с результатами натурных испытаний. Если уклонения расчетных значений от тех, что получены в натурных экспериментах, выходят за пределы точности наблюдений, то говорят, что модель неадекватна, и не может быть принята. В этом случае следует вернуться к первому этапу и уточнить математическую модель. Возможно, учесть факторы и связи, ранее не отраженные в модели.

Модель признается адекватной, если степень её соответствия характеру моделируемого явления не выходит за пределы точности наблюдений.

В англоязычной научной литературе существует термин «computer simulation» или «компьютерная имитация», что означает реализованную на ЭВМ математическую модель, оперирующую системой динамических уравнений.

4 этап. Проведение серии вычислительных экспериментов на основе модели, признанной адекватной, при различных значениях параметров, характеризующих изучаемый объект и, возможно, разных начальных условиях. На основе данных этих экспериментов исследователи получают новые знания об объекте и его свойствах, находят оптимальные параметры и режимы работы.

5 этап. Анализ модели в связи с накоплением новых научных данных об изучаемых объектах и модернизация модели на их основе.

Контрольные вопросы.

1) Что такое модель? Приведите примеры моделей.

2) Назовите известные Вам способы классификации моделей.

3) На какие два класса можно разбить модели по способу представления? Приведите примеры.

4) Что такое математическая модель? Как её можно классифицировать?

5) Что подразумевают под компьютерным цифровым моделированием?

6) Назовите этапы компьютерного моделирования.

7) Что является целью компьютерного моделирования?

2. Получение математических моделей электротехнических цепей

Компьютерное моделирование электротехнических систем связано с использованием физических законов, описывающих электрические, а также электромагнитные явления и процессы.

Математическая модель любого объекта характеризуются внутренними, внешними, выходными параметрами и фазовыми переменными. Внутренние параметры модели в нашем случае определяются номиналами элементов принципиальной схемы. Если проектируемый объект содержит n элементарных компонентов, то и его математическая модель будет определяться параметрами, которые образуют вектор внутренних параметров $W = |w_1 ... w_n|^T$.

Выходные параметры модели могут характеризовать функциональные, эксплуатационные, конструкторско-технологические, экономические и другие характеристики проектируемого объекта. К таким показателям могут относиться коэффициенты передачи, сила тока на участке цепи, надежность, стоимость и т.п. Вектор выходных параметров модели будем обозначать $Y=| y_1 \quad y_k |^T$.

Внешние параметры модели — это характеристики внешней по отношению к проектируемому объекту среды, а также рабочие управляющие воздействия. Будем обозначать вектор внешних параметров $X=| x_1 \quad x_k |^T$.

В электрических цепях мы будем рассматривать в качестве выходных параметров модели либо напряжения, либо токи, либо заряд конденсатора, в качестве входных параметров – напряжение, подаваемое на вход цепи (напряжение, создаваемое источником э.д.с.).

2.1. Математические модели неразветвленных цепей

2.1.1. Математическая модель RC-цепи. Пример 1

Начнем рассмотрение вопроса получения математических моделей с примера цепи, содержащей всего два последовательно

соединенных элемента: резистор номинального сопротивления R и конденсатор электрической ёмкости C (рис.1).

Для построения математической модели примем допущение, что величины R и C являются постоянными в течение всего времени наблюдения. Такой приём физической и математической идеализации используется в случаях, когда цель моделирования позволяет пренебречь возможными зависимостями сопротивлений и ёмкостей от температурных режимов эксплуатации и речь не идет об элементах цепи, управляемых током или напряжением (в таких случаях математическая модель превращается в дифференциальное уравнение с переменными коэффициентами).

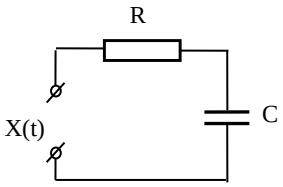


Рис. 1. RC - цепь

В зависимости от того, какой параметр будет выбран в качестве выходного, можно получать различные математические модели.

Рассмотрим в качестве выходного параметра заряд конденсатора q(t), который является непрерывной функцией времени. Получим математическую модель нашей цепи при входном напряжении 1 В.

Математическую модель объекта, представляющего объединение компонентов, получают на основе математических моделей этих компонентов. В нашей схеме для резистора справедлив закон Ома (1), выражающий соотношение между током и напряжением:

$$U_R(t) = R \cdot I_R(t). \quad (1)$$

Известно соотношение (2), описывающее связь между зарядом конденсатора и напряжением на нем:

$$q_C(t) = C \cdot U_C(t). \quad (2)$$

Объединение компонентных уравнений в математическую модель объекта осуществляется на основе фундаментальных физических законов, выражающих условия непрерывности и равновесия фазовых переменных, например законов (правил) Кирхгофа (3). Уравнения, описывающие эти законы, называют топологическими; они отражают связи между компонентами в устройстве.

$$U_R(t) + U_C(t) = X(t). \quad (3)$$

Получим из соотношения (2) U_C и подставим его, а также выражение (1), в выражение (3):

$$R \cdot I_R(t) + \frac{q_C(t)}{C} = X(t). \quad (4)$$

Полученное выражение не может быть признано математической моделью, так как решение этого уравнения получить невозможно: оно содержит две неизвестные функции $I_R(t)$ и $q_C(t)$.

Известно, что токи на различных элементах цепи при их последовательном соединении равны, то есть

$$I_C(t) = I_R(t). \quad (5)$$

Продифференцировав выражение (2), получим:

$$I_C(t) = \frac{dq_C(t)}{dt} = C \cdot \frac{dU_C(t)}{dt}. \quad (6)$$

Используя это выражение и соотношение (4), получим искомую математическую модель:

$$R \cdot \frac{dq_C(t)}{dt} + \frac{q_C(t)}{C} = X(t). \quad (7)$$

Математическая модель содержит внутренние (R и C) и внешние X(t) параметры цепи, а также одну неизвестную функцию $q_C(t)$ и её производную.

Таким образом, мы получили математическую модель в виде дифференциального уравнения первого порядка.

2.1.2. Математическая модель RC-цепи. Пример 2

Выберем теперь в качестве выходного параметра напряжение на конденсаторе $U_C(t)$.

Используя соотношения (2) и (7), получим математическую модель в виде:

$$R \cdot C \cdot \frac{dU_C(t)}{dt} + U_C(t) = X(t).$$

Преобразуем это выражение, приведя его к следующему виду:

$$\frac{dU_C(t)}{dt} = \frac{X(t) - U_C(t)}{R \cdot C}. \quad (8)$$

Если записать это дифференциальное уравнение, применив преобразование Лапласа, широко используемое в электротехнике и теории автоматического управления, получим

$$R \cdot C \cdot p \cdot U_C(t) + U_C(t) = X(t).$$

Записав передаточную функцию звена, представленного данной схемой, получим:

$$W(t) = \frac{U_C(t)}{X(t)} = \frac{1}{R \cdot C \cdot p + 1}.$$

Это передаточная функция апериодического звена первого порядка. Постоянная времени, характеризующая инерционность этого звена, $T = R \cdot C.$

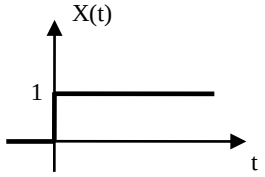


Рис. 2. Единичная ступенчатая функция.

Если на вход последовательной RC-цепи (рис. 1) с полностью разряженным конденсатором подать постоянное напряжение в виде единичной ступенчатой функции, или функции Хевисайда (рис. 2), то напряжение на конденсаторе достигнет значения ≈ 1 за время, равное $3 \cdot T$, то есть $3 \cdot R \cdot C.$

Проинтегрировав выражение (8), получим

$$U_C(t) = \frac{1}{R \cdot C} \int (X(t) - U_C(t))\, dt.$$

2.1.3. Математическая модель RC-цепи. Пример 3

Пусть теперь выходным параметром будет напряжение на резисторе $U_R(t)$. Воспользуемся равенством токов (5), подставив в него выражение для тока на резисторе, полученное из (1) и на конденсаторе (6):

$$\frac{U_R(t)}{R} = C \cdot \frac{dU_C(t)}{dt}. \qquad 9)$$

Дальнейшие действия сводятся к тому, чтобы из 2-го правила Кирхгофа исключить неизвестные функции, не являющиеся выходными параметрами модели, то есть, $U_C(t)$. Следует отметить, что из выражения (6) можно выразить не саму неизвестную функцию $U_C(t)$, а её производную. А для того, чтобы эту производную можно было подставить во второе правило Кирхгофа (3), его необходимо продифференцировать:

$$\frac{dU_R(t)}{dt} + \frac{dU_C(t)}{dt} = \frac{dX(t)}{dt}. \qquad 10)$$

Подставив сюда выражение для производной функции $U_C(t)$ из (9), получим:

$$\frac{dU_R(t)}{dt} + \frac{U_R(t)}{R \cdot C} = \frac{dX(t)}{dt}. \qquad 11)$$

Уравнение содержит одну неизвестную функцию $U_R(t)$ и ее производную, производную от входного напряжения и параметры цепи. Таким образом, получена третья математическая модель для одной и той же электрической цепи.

2.2.1. RLC-цепь (колебательный контур). Пример 1

Рассмотрим электрическую схему, состоящую из последовательно соединённых резистора, конденсатора и катушки, которую принято называть колебательным контуром.

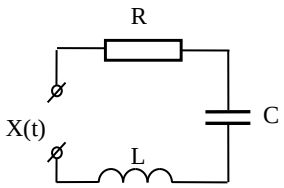


Рис. 3. RLC - цепь

Для этой цепи также можно получать разные математические модели, если выбирать в качестве выходных разные параметры. Например, можно в качестве выходных параметров рассматривать такие функции времени, как напряжение на резисторе, или на конденсаторе, или на катушке и т.д.

Рассмотрим математическую модель, для которой выходным параметром будет напряжение на резисторе.

Основные физические законы, используемые для вывода модели, это закон Ома (1), связь между зарядом конденсатора и напряжением на нём (2), и соотношение, справедливое для катушки:

$$U_L(t) = L \cdot \frac{di_L(t)}{dt}. \qquad 12)$$

Связь между элементами цепи выражается вторым законом Кирхгофа:

$$U_R(t) + U_C(t). + U_L(t) = X(t) \qquad 13)$$

Задача получения математической модели в виде дифференциального уравнения сводится в нашем случае к исключению из выражения (13) неизвестных функций, не являющихся выходным напряжением $U_R(t)$. Ключом к решению этой задачи является равенство токов:

$$I_C(t) = I_R(t) = I_L(t). \qquad 14)$$

Из первой части выражения (14) получим

$$C \cdot \frac{dU_C(t)}{dt} = \frac{U_R(t)}{R}. \qquad 15)$$

Из второй части выражения (14) невозможно исключить ток $I_L(t)$, так как в выражении (12) мы имеем производную от тока. Поэтому продифференцируем вторую часть равенства токов

$$\frac{dI_R(t)}{dt} = \frac{dI_L(t)}{dt} \text{ и} \qquad 16)$$

подставим сюда производную от тока на катушке из (12) и продифференцированный закон Ома (1):

$$\frac{1}{R} \cdot \frac{dU_R(t)}{dt} = \frac{U_L(t)}{L}. \qquad 17)$$

Теперь необходимо исключить из 2-го правила Кирхгофа неизвестные функции, не являющиеся выходными параметрами модели, то есть, $U_C(t)$ и $U_L(t)$. Надо отметить, что в выражении (15) содержится производная от напряжения на конденсаторе, а не сама эта функция. Это значит, что нужно будет продифференцировать второе правило Кирхгофа (13)

$$\frac{dU_R(t)}{dt}+\frac{dU_C(t)}{dt}+\frac{dU_L(t)}{dt}=\frac{dX(t)}{dt}, \text{ а} \qquad 18)$$

также выражение (17), так как теперь потребуется еще и производная от напряжения на катушке:

$$\frac{dU_L(t)}{dt}=\frac{L}{R}\cdot\frac{d^2U_R(t)}{dt^2}. \qquad 19)$$

Выполнив подстановку (15) и (19) в (18) получим математическую модель:

$$\frac{L}{R}\frac{d^2U_R(t)}{dt^2}+\frac{dU_R(t)}{dt}+\frac{1}{R\cdot C}U_R(t)=\frac{dX(t)}{dt}. \qquad 20)$$

2.2.2. RLC-цепь. Пример 2

Рассмотрим ту же самую схему (рис. 3), состоящую из последовательно соединённых резистора, конденсатора и катушки.

Составим математическую модель, для которой выходным параметром будет напряжение на конденсаторе.

По-прежнему, основной задачей при построении математической модели будет исключение неизвестных функций из второго правила Кирхгофа.

$$U_R(t)+U_C(t).+U_L(t)=X(t)$$

Такими функциями будут в нашем случае зависимости напряжения на резисторе и на катушке от времени.

Исключить их из этого выражения можно, пользуясь тем, что токи на различных элементах цепи одинаковы. Так как

$$I_C(t)=I_R(t),$$

то на основе выражения (6) можно записать, что

$$C\cdot\frac{dU_C(t)}{dt}=\frac{U_R(t)}{R}. \qquad (21)$$

Также равны токи на конденсаторе и катушке:

$$I_C(t)=I_L(t).$$

Для того, чтобы перейти к выражению, содержащему напряжения, потребуется продифференцировать это равенство:

$$\frac{dI_C(t)}{dt} = \frac{dI_L(t)}{dt}. \qquad (22)$$

Выразим необходимые производные из выражений (12) и (6):

$$\frac{dI_C(t)}{dt} = C\frac{d^2U_C(t)}{dt^2}, \quad \frac{dI_L(t)}{dt} = \frac{U_L(t)}{L}.$$

Подставив их в (22), получим

$$C\frac{d^2U_C(t)}{dt^2} = \frac{U_L(t)}{L}.$$

Подставив выражения для неизвестных функций из (21) и (22) во второе правило Кирхгофа, получим математическую модель интегрирующей RLC-цепи.

$$CL\frac{d^2U_C(t)}{dt^2} + R\,C\frac{dU_C}{dt} + U_C(t) = X(t) \qquad (23)$$

2.2.3. RLC-цепь. Построение математической модели с использованием преобразования Лапласа

Есть другой способ получить математическую модель, используя преобразование Лапласа.

Для этого перепишем выражения для тока на конденсаторе и для производной тока на катушке в преобразованиях Лапласа:

$$I_C = CpU_C(t)\ , \quad pI_L = \frac{U_L(t)}{L}$$

Так как токи на конденсаторе и катушке равны, то

$$pI_C = pI_L\ , \quad Cp^2U_C(t) = \frac{U_L(t)}{L}$$

Отсюда можно выразить напряжение на катушке через вторую производную от напряжения на конденсаторе

$$U_L(t) = CLp^2U_C(t). \qquad (24)$$

Далее нужно выразить напряжение на резисторе через напряжение на конденсаторе. Это можно сделать, учитывая равенство токов:

$$U_R(t) = RCpU_C(t) \qquad (25)$$

Подставим выражения (24) и (25) во второе правило Кирхгофа:

$$CLp^2U_C(t) + RCpU_C(t) + U_C(t) = X(t) \qquad (26)$$

Заменив символ p в соответствующей степени нужной производной, получим ту же модель:

$$CL\frac{d^2U_C(t)}{dt^2} + RC\frac{dU_C}{dt} + U_C(t) = X(t)$$

2.3. Задача автоматического подбора параметров цепи

При проектировании различных электро- и радиотехнических устройств важно обеспечить определенные параметры их функционирования, которые соответствовали бы техническому заданию. При наличии математической модели проектируемого устройства одним из способов решения такой задачи может быть автоматический подбор параметров, осуществляемый при помощи компьютера.

Рассмотрим простейший вариант задачи автоматического подбора параметров дифференцирующей RC-цепи, на вход которой подается напряжение X(t) в виде единичной ступенчатой функции, или функции Хевисайда.

Суть задачи состоит в том, чтобы подобрать с помощью компьютерного моделирования один из параметров цепи (R или C) так, чтобы на выходе цепи за некоторое конечное время tk установилось напряжение, равное по величине Utr.

Первое, что необходимо выяснить при решении этой задачи: изменение какого из параметров оказывает более существенное влияние на выходное напряжение.

В случае RC-цепи (рис. 1) таким параметром оказывается ёмкость конденсатора C. Чтобы убедиться в этом, достаточно выполнить три компьютерных эксперимента. Первый – выбрав конкретные значения ёмкости конденсатора C, сопротивления резистора R, задав единичное входное напряжение и установив некоторое конечное время наблюдения и шаг ведения расчёта. Второй эксперимент проводится при изменении одного единственного параметра, например, при увеличенной в 2 раза ёмкости конденсатора C. Третий – при увеличенном в 2 раза сопротивлении резистора R. Во всех трёх случаях фиксируются значения выходного напряжения в конечный

момент времени. Их сравнение позволит понять, какой из параметров схемы наиболее существенно влияет на выходное напряжение, а также понять логику достижения цели.

Теперь обсудим, как составить алгоритм автоматического подбора параметра. Очевидно, что необходимо задать какое-то начальное значение ёмкости. Назовём его C0. Также следует задать начальное значение шага hC, с которым будет подбираться требуемое значение C.

Также следует задать точность ε, с которой мы будем оценивать, достигнуто ли при текущем значении ёмкости C необходимое значение выходного напряжения Utr с допустимой погрешностью.

Примерную схему решения задачи смотрите на рисунке 4.

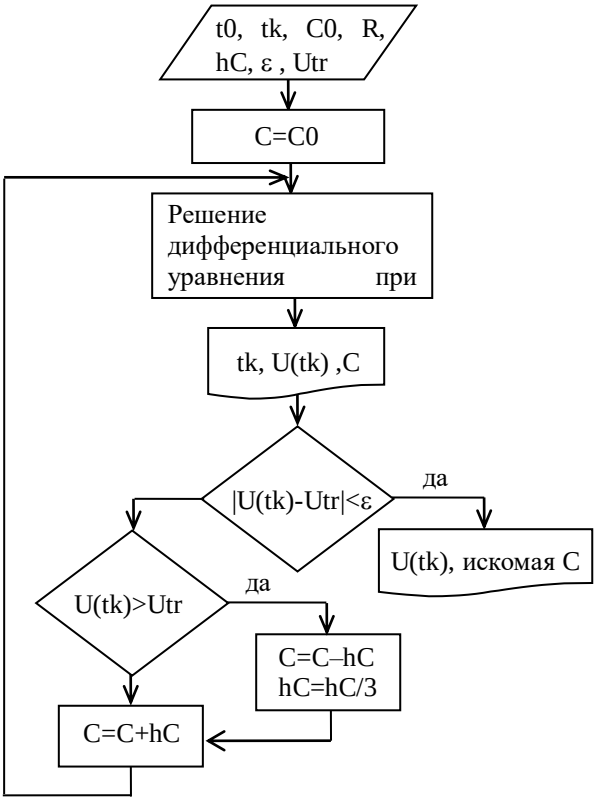


Рис. 4. Схема алгоритма автоподбора ёмкости конденсатора C.

Ниже смотрите листинг программы, решающей задачу автоподбора емкости конденсатора C с построением графиков получаемых решений.

```
#define USE MATH DEFINES
#include "graphics.h"
#include <math.h>
#include <stdio.h>
//объявление глобальных переменных
double xmin,xmax,ymin,ymax;
//функция пересчета математической
//координаты x в физическую (пиксель)
double sx(double x)
{
  return(x-xmin)/(xmax-xmin)*getmaxx();
}
//функция пересчета математической
//координаты y в физическую (пиксель)
double sy(double y)
{
return getmaxy()-(y-ymin)/(ymax-ymin)*getmaxy();
}
//функция, вычисляющая правую часть
//дифференциального уравнения
double f(double x,double y,double R,double C)
 {
  return -y/(R*C);
 }
//"главная" функция
int main(void)
{
  double x,y,h,y0,k1,k2,k3,k4,z,R,C,a,HC,UTR=0.2;
  double mx[10000],my[10000];
  int n,i;
//ввод начального и конечного времени (xmin, xmax)
//и шага изменения времени h
  printf("Input the range and step:xmin,xmax,h\n")
  scanf("%lf,%lf,%lf",&xmin,&xmax,&h);
//ввод параметров электрической цепи
  printf("Input R,C\n");
  scanf("%lf,%lf",&R,&C);
//ввод начального значения функции y0
  printf("Input beginning function value y0\n");
```

```
  scanf("%lf",&y0);
  y0=1;xmin=0;xmax=0.25;h=0.001;R=20000;C=0.000004;
//открытие графического окна
  initwindow(400,200);
  setcolor(WHITE);
  setlinestyle(SOLID_LINE,0,NORM_WIDTH);
//рисование осей: горизонтальной и вертикальной
  line(sx(xmin),sy(0),sx(xmax),sy(0));
  line(sx(0),sy(ymin),sx(0),sy(1));
//подготовка к рисованию графика функции
  setcolor(RED);
  setlinestyle(SOLID_LINE,0,THICK_WIDTH);
//значение шага изменения ёмкости
  HC=C/3;
//количество точек на интервале
  n=(xmax-xmin)/h;
//подготовка к определению минимального
//и максимального значений решения
  ymin=y0;
  ymax=ymin;
//цикл,который работает, если расчётное
//значение напряжения y по модулю отличается
//от требуемого UTR более, чем на 0.0001
  while(fabs(y-UTR)>0.0001)
   {
      y=y0;
      x=xmin;
//цикл решения дифференциального уравнения
//при конкретном текущем значении C
      for(i=0;i<=n;i++)
      {
        mx[i]=x;
        my[i]=y;
        if(y>ymax)ymax=y;
        if(y<ymin)ymin=y;
//метод Рунге-Кутта
      k1=h*f(x,y,R,C);
      k2=h*f(x+h/2, y+k1/2,R,C);
      k3=h*f(x+h/2, y+k2/2,R,C);
```

```
      k4=h*f(x+h, y+k3,R,C);
//значение решения в следующий
//момент времени
      y=y+(k1+2*k2+2*k3+k4)/6;
      x=x+h;
    }
//прорисовываем решение при
//конкретном значении C
x=mx[0];
y=my[0];
moveto(sx(x),sy(y));
  for(i=1;i<=n;i++)
      {
            x=mx[i];
            y=my[i];
            lineto(sx(x),sy(y));
      }
//реализация алгоритма автоподбора
  if(y>UTR)
  {
       C=C-HC;
       HC=HC/3;
  }
  C=C+HC;
//распечатка выходного напряжения
//в конечный момент времени и текущего C
  printf("t=%lf Uvux=%lf C=%.9lf\n",mx[n],my[n],C);
  delay(800);
      }
printf("podobrali C=%.9lf\n",C);
getch();
closegraph();
return 0;
}
```

Результаты работы программы представлены в виде двух окон, одно из которых содержит значения выходного напряжения в конечный момент времени при различных значениях емкости конденсатора C (рис. 5), а другое – соответствующие графики решений (рис. 6).

```
Input the range and step:xmin,xmax,h
0,0.5,0.01
Input R,C
10000,0.000003
Input beginning function value y0
0
t=0.250000 Uvux=0.043937 C=0.000005333
t=0.250000 Uvux=0.095967 C=0.000006667
t=0.250000 Uvux=0.153355 C=0.000008000
t=0.250000 Uvux=0.209611 C=0.000007111
t=0.250000 Uvux=0.172422 C=0.000007556
t=0.250000 Uvux=0.191204 C=0.000008000
t=0.250000 Uvux=0.209611 C=0.000007704
t=0.250000 Uvux=0.197386 C=0.000007852
t=0.250000 Uvux=0.203522 C=0.000007753
t=0.250000 Uvux=0.199436 C=0.000007802
t=0.250000 Uvux=0.201482 C=0.000007770
t=0.250000 Uvux=0.200119 C=0.000007759
t=0.250000 Uvux=0.199664 C=0.000007764
t=0.250000 Uvux=0.199891 C=0.000007770
t=0.250000 Uvux=0.200119 C=0.000007766
t=0.250000 Uvux=0.199967 C=0.000007768
podobrali C=0.000007768
```

Рис. 5. Численные результаты работы программы автоподбора.

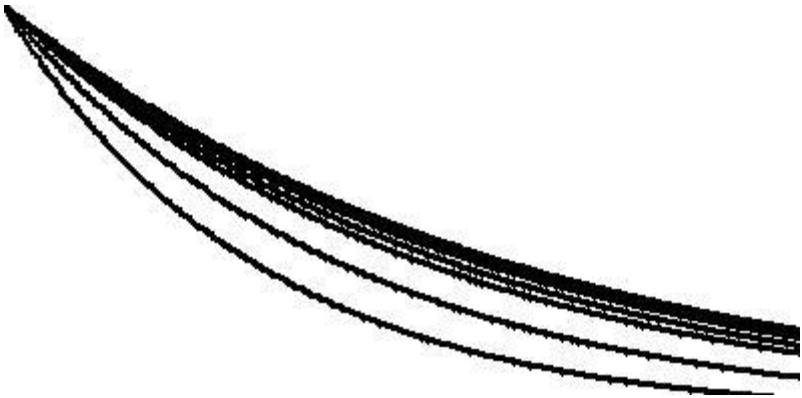


Рис. 6. Графики решений при различных значениях емкости конденсатора C.

2.4. Математические модели разветвленных цепей.

2.4.1. Математическая модель разветвлённой RC-цепи.

Рассмотрим разветвленную электрическую схему, состоящую из двух резисторов и конденсатора (рис. 7).

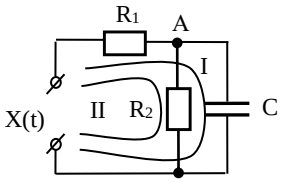


Рис. 7. Пример разветвленной RC - цепи

Получим математическую модель, в которой неизвестной функцией времени будет напряжение на конденсаторе.

Основные физические законы, используемые для вывода модели, это закон Ома (1), а также связь между зарядом конденсатора и напряжением на нём (2).

Запишем для контуров I и II второй закон Кирхгофа:

$$U_{R1}(t) + U_C(t) = X(t) \qquad (27)$$
$$U_{R1}(t) + U_{R2}(t) = X(t)$$

Сравнивая полученные выражения, находим подтверждение того факта, что напряжения на параллельно соединенных элементах цепи равны:

$$U_{R2}(t) = U_C(t) \qquad (28)$$

Задача получения математической модели сводится к исключению из выражения (27) неизвестной функции $U_{R1}(t)$. Для этого запишем первый закон Кирхгофа относительно узла А:

$$I_{R1}(t) = I_{R2}(t) + I_C(t) \qquad (29)$$

Выразим токи в выражении (29) через напряжения

$$\frac{U_{R1}(t)}{R1} = \frac{U_{R2}(t)}{R2} + C\frac{dU_C(t)}{dt} \qquad (30)$$

Получим отсюда $U_{R1}(t)$ с учетом (28):

$$U_{R1}(t) = \frac{R1}{R2}U_C(t) + CR1\frac{dU_C(t)}{dt} \qquad (31)$$

Теперь подставим выражение (31) в выражение (27) и получим математическую модель разветвленной цепи (рис.7):

$$CR1\frac{dU_C(t)}{dt} + \left(\frac{R1}{R2} + 1\right)U_C(t) = X(t) \qquad (32)$$

2.5. Математические модели электрических цепей с нелинейностями. Однополупериодный выпрямитель

Математические модели рассмотренных ранее электрических цепей представляют собой линейные дифференциальные уравнения. Их решение может легко быть получено не только численными методами, но и аналитически. Однако огромное количество динамических систем являются по природе своей нелинейными и описываются нелинейными дифференциальными уравнениями. Нелинейности часто представляют собой кусочно-линейные или разрывные характеристики, которые не могут быть представлены в виде одного аналитического выражения.

В качестве конкретного примера рассмотрим описание процессов в схеме однополупериодного выпрямителя, нагруженного активным сопротивлением (рис. 8).

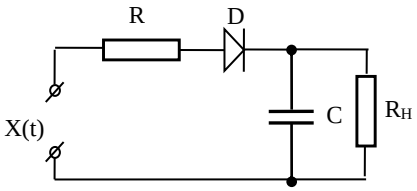


Рис. 8. Однополупериодный выпрямитель.

Нелинейная характеристика диода в упрощенном, идеализированном, виде может быть описана с помощью единичной функции Хевисайда:

$$i_D = f(U_D) = \frac{U_D(t)}{R_D} \cdot \mathbf{1}(U_D), \quad \text{где } \mathbf{1}(U_D) = \begin{cases} 0, \text{если } U_D < 0 \\ 1, \text{если } U_D \geq 0 \end{cases} \qquad (33)$$

Соотношения, справедливые для линейных элементов нам известны:

$$i_R = \frac{U_R}{R}, \qquad i_C = C \cdot \frac{dU_C(t)}{dt}, \qquad i_H = \frac{U_{R_H}}{R_H} \qquad (34)$$

Согласно второму закону Кирхгофа получим соотношение для напряжений:

$$U_R(t) + U_D(t). + U_{R_H}(t) = X(t)\,;\; U_{R_H}(t) = X(t) - U_R(t) - U_D(t) \qquad (35)$$

Согласно первому закону Кирхгофа получим соотношение для токов:

$$i_D = i_C + i_{R_H} \tag{36}$$

Получим математическую модель, описывающую процессы в выпрямителе и выражающую зависимость выходного напряжения $U_{R_H}(t)$ от входного $X(t)$.

В силу того, что резистор и диод соединены последовательно,

$$I_R(t) = I_D(t) \quad \text{и} \quad f(U_D) = \frac{U_R(t)}{R} \tag{37}$$

Так как конденсатор подключен параллельно нагрузке

$$U_C = U_{R_H} \tag{38}$$

Заменим токи в первом законе Кирхгофа (36) на соответствующие напряжения и их производные с учетом (34) и (38):

$$f(U_D) = C \cdot \frac{dU_{R_H}}{dt} + \frac{U_{R_H}}{R_H} \tag{39}$$

Нашей целью является исключение из второго закона Кирхгофа неизвестных функций $U_R(t)$ и $U_D(t)$. Но в данном случае сделать это затруднительно, так как $U_D(t)$ является аргументом нелинейной характеристики диода (33). Поэтому в математической модели в качестве неизвестной функции будем рассматривать $U_D(t)$.

Рассчитав из математической модели $U_D(t)$ в каждый момент времени и подставив его в выражение (35), мы найдем выходное напряжение $U_{R_H}(t)$.

Итак, получим зависимость между входным напряжением и напряжением на диоде. Продифференцируем выражение (35):

$$\frac{dU_{R_H}(t)}{dt} = \frac{dX(t)}{dt} - \frac{dU_R(t)}{dt} - \frac{dU_D(t)}{dt} \tag{40}$$

Перепишем (40) с учетом (33) и (37):

$$\frac{dU_{R_H}(t)}{dt} = \frac{dX(t)}{dt} - \left(1 + \frac{R}{R_D} \cdot \mathbf{1}(U_D)\right) \cdot \frac{dU_D(t)}{dt} \tag{41}$$

Подставим в уравнение (39) функцию $U_{R_H}(t)$ и ее производную из выражений (35) и (41):

$$f(U_D) = C\left(\frac{dX}{dt} - \left(1 + \frac{R}{R_D}\mathbf{1}(U_D)\right)\frac{dU_D}{dt}\right) + \frac{1}{R_H}\left(X - Rf\left(U_D\right) - U_D\right) \quad (42)$$

Разрешим полученное уравнение относительно производной искомой функции $U_D(t)$, то есть представим его в форме Коши, а также подставим в получаемое выражение нелинейную характеристику диода $f(U_D)$ из (33):

$$\frac{dU_D}{dt} = \frac{\frac{dX}{dt} + \frac{X - U_D}{R_H \cdot C} - \frac{U_D}{R_D \cdot C} \cdot \left(1 + \frac{R}{R_D}\right) \cdot \mathbf{1}(U_D)}{1 + \frac{R}{R_D} \cdot \mathbf{1}(U_D)} \quad (43)$$

На вход выпрямителя обычно подается синусоидальное напряжение:

$$X(t) = A \cdot \sin(\omega \cdot t).$$

Выходное напряжение, получаемое на нагрузке, зависит от параметров входного напряжения, параметров элементов схемы самого выпрямителя и нагрузки и может иметь вид, представленный на рис. 9.

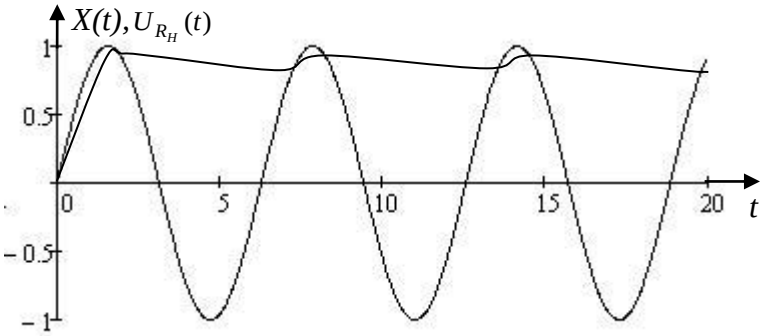


Рис. 9. Напряжение на выходе однополупериодного выпрямителя.

Время интегрирования следует выбирать достаточным для наблюдения не только переходного процесса, но и установившегося режима.

2.6. Моделирование электрических цепей в пакете MathCAD 14

В пакете MathCAD существуют несколько способов решения дифференциальных уравнений, являющихся математическими моделями динамических систем. Рассмотрим некоторые из способов.

В качестве **первого примера** рассмотрим решение в MathCADe математической модели (8) цепи, рассмотренной в п. 2.1.1.

В примере использован блок решения Given для формулировки задачи – задания дифференциального уравнения и начальных условий в момент времени t0=0, а также решающая функция Odesolve. Параметры цепи следует задавать до блока решения, так как
использование в нем команды Definition (обозначаемой знаком :=) запрещено.

$R := 20000 \qquad C := 0.000001 \qquad X(t) := 1$

Given

$$\frac{d}{dt}Uc(t) = \frac{X(t) - Uc(t)}{R \cdot C}$$

$Uc(0) = 0$

$Uc := Odesolve(t, 0.2)$

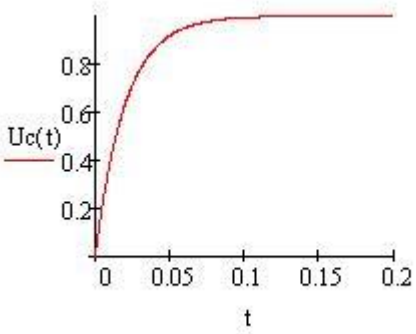


Рис. 10. График зависимости напряжения на конденсаторе от времени.

Замечания. Для ввода производной следует использовать шаблон с панели инструментов Calculus. Первый параметр функции Odesolve – независимая переменная (время), второй – конечное время наблюдения.

Второй пример – решение математической модели (8) цепи, рассмотренной в п. 2.1.1. с помощью функции rkfixed (метод Рунге-Кутта с фиксированным шагом).

Параметры цепи: $R := 20000 \quad C := 0.000001$

Входной сигнал: $X(t) := 1$ Конечное время $tk := 0.2$

Количество точек решения на интервале $[to, tk]$ $N := 1000$

Необходимо решить дифференциальное уравнение:

$\frac{d}{dt}Uc(t) = f(t, Uc)$, где

$$f(t, Uc) := \frac{X(t) - Uc}{R \cdot C}$$

Начальные условия: $Uc0 := 0$ $t0 := 0$

Для получения решения нужно задать вектор начальных условий:

$nu_0 := Uc0$

Задаем функцию, равную правой части дифференциального уравнения, второй аргумент которой является вектором искомых значений функции Uc:

$D(t, Uc) := f(t, Uc_0)$

В качестве решения получаем матрицу U:

$U := rkfixed(nu, t0, tk, N, D)$

$T := U^{\langle 0 \rangle}$ - вектор - столбец значений времени

$Uc := U^{\langle 1 \rangle}$ - вектор - столбец значений напряжения на конденсаторе

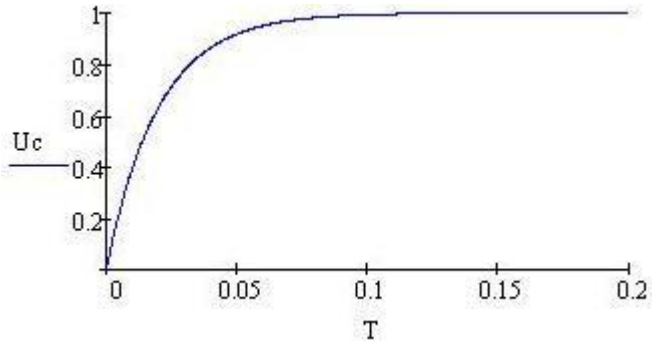


Рис. 11. График зависимости напряжения на конденсаторе от времени, полученный с помощью решающей функции rkfixed.

Замечание. Эту задачу можно решить в MathCADe и с помощью других функций, параметры которых аналогичны параметрам функции rkfixed. Такой функцией, например, является функция Rkadapt, реализующая метод Рунге-Кутта с адаптивным подбором шага. Эта функция позволяет получать решение с мелким шагом в областях быстрого изменения искомой функции и с более крупным шагом там, где решение меняется медленно, что и повышает точность, и сокращает время, требуемое для решения уравнения.

Также может быть применена функция, решающая дифференциальные уравнения методом Адамса (Adams), или функция Radau, реализующая метод Radau 5. В случаях, когда известно, что решение является гладкой функцией, имеет смысл использовать функцию Bulstoer, решающую дифференциальные уравнения методом Булирша-Штёра.

Третий пример – решение математической модели колебательного контура из п. 2.2.1.

В решении также использована функция Odesolve. Так как модель представляет собой дифференциальное уравнение второго порядка, в блоке Given задано два начальных условия – значение искомой функции, а также её производной в момент времени t0=0.

$R := 500000 \qquad C := 0.000001 \qquad L := 0.004 \qquad X(t) := \sin(2 \cdot t)$

Given

$$\frac{L}{R}\frac{d^2}{dt^2}Ur(t) + \frac{d}{dt}Ur(t) + \frac{1}{R \cdot C}Ur(t) = \frac{d}{dt}X(t)$$

$Ur(0) = 0$

$Ur'(0) = 0$

$Ur := Odesolve(t, 20)$

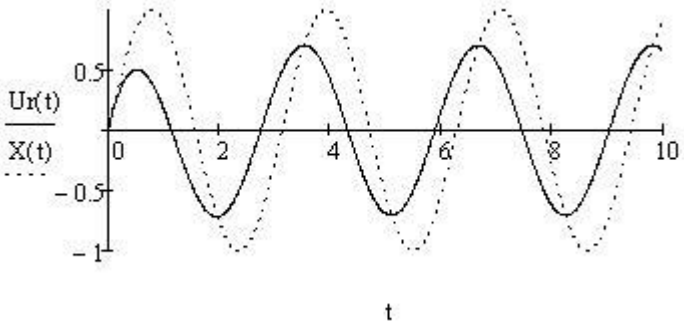


Рис. 12. Графики входного X(t) и выходного Ur(t) напряжений на резисторе RLC-цепи (колебательного контура).

Четвертый пример – решение этой же задачи в параметрическом виде. Здесь решение запрашивается в виде параметрической функции $fUr(R)$, зависящей не только от времени, но и от одного из параметров самой цепи, в данном примере – от величины сопротивления резистора R.

После получения параметрического решения в общем виде с помощью решающей функции Odesolve, находятся решения при значениях R=500 кОм и R=50 кОм. Эти решения записываются в новые функции, имена которых Ur500000 и Ur50000. Графики этих функций изображены на рисунке 13.

$R := 500000$ $C := 0.000001$ $L := 0.004$ $X(t) := \sin(2 \cdot t)$

$t0 := 0$

Given

$$\frac{L}{R} \cdot \frac{d^2}{dt^2} Ur(t) + \frac{d}{dt} Ur(t) + \frac{1}{R \cdot C} \cdot Ur(t) = \frac{d}{dt} X(t)$$

$Ur(t0) = 0$

$Ur'(t0) = 0$

$fUr(R) := Odesolve(t, 40)$

$Ur500000 := fUr(500000)$ $Ur50000 := fUr(50000)$

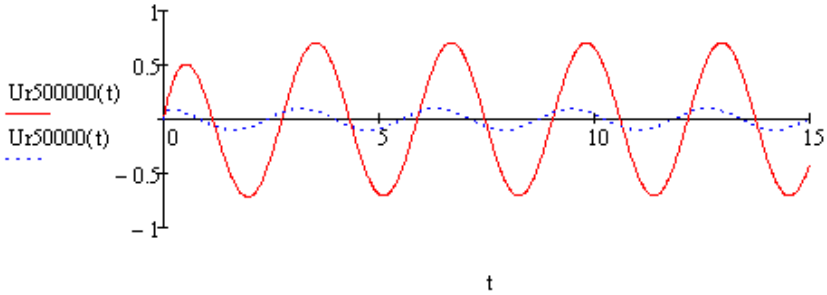


Рис. 13. Графики выходного напряжения на резисторе RLC-цепи при разных значениях сопротивления R.

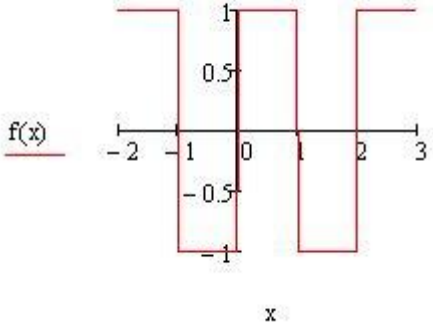
2.7. Моделирование электрических сигналов в пакете MathCAD 14.

Сигналы, подаваемые на вход электрической цепи, могут быть весьма разнообразными. Для компьютерного моделирования этих сигналов могут использоваться комбинации стандартных математических функций пакета MathCAD, а также известные математические методы, например, разложение произвольной математической функции в ряд Фурье.

Рассмотрим несколько примеров моделирования.

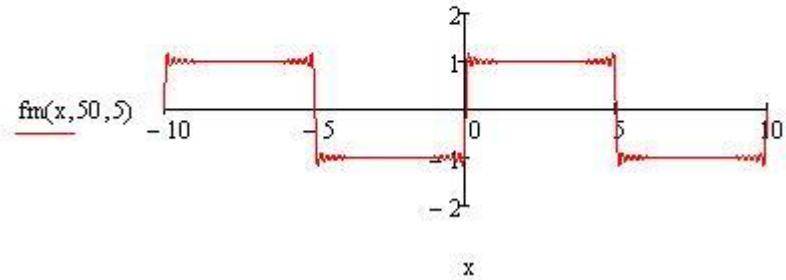
Пример 1. Пусть требуется смоделировать сигнал «меандр». Это можно сделать, используя функцию MathCAD sign(x), которая принимает значение 1, если аргумент x>0, значение -1, если x<=0 и равна 0 при x=0.

$$f(x) := \mathrm{sign}(\sin(x \cdot \pi))$$



Другой вариант построения меандра – с помощью ряда Фурье:

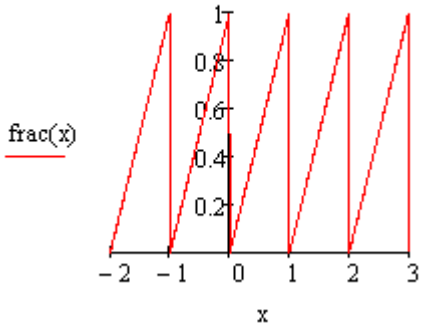
$$fm(x, N, T) := \frac{4}{\pi} \cdot \left[\sum_{n=1}^{N} \left(\frac{1}{n} \cdot \sin\left(\frac{n \cdot \pi \cdot x}{T} \right) \cdot \mathrm{mod}(n, 2) \right) \right]$$



Здесь применена функция mod(n,2), которая возвращает в качестве результата остаток от деления n на 2, то есть равна 0 для четных и 1 для нечетных значений n.

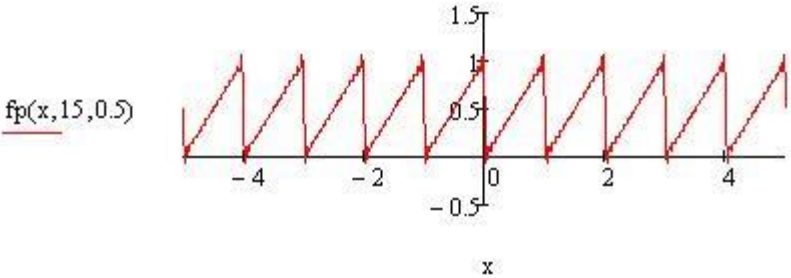
Пример 2. Пусть требуется смоделировать пилообразный сигнал. Для его получения используется функция floor(x), результат вычисления которой – наибольшее целое число, меньшее либо равное x.

$$frac(x) := x - floor(x)$$



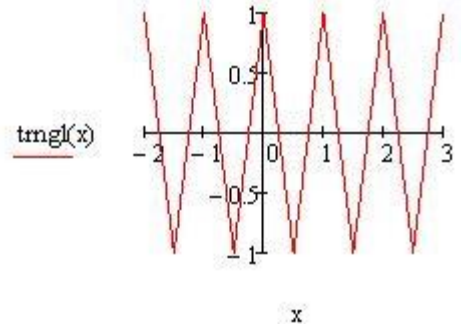
Аналогичный сигнал, смоделированный с помощью ряда Фурье:

$$fp(x, N, T) := \frac{1}{2} - \frac{1}{\pi} \cdot \left[\sum^{N} \left(\frac{1}{n} \cdot \sin\left(\frac{n \cdot \pi \cdot x}{T} \right) \right) \right]$$



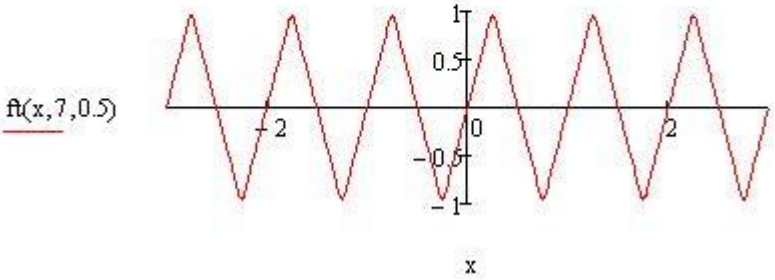
Пример 3. Смоделируем треугольный сигнал, применив функцию frac(x), заданную в примере 2.

$$trngl(x) := |frac(x) \cdot 2 - 1| \cdot 2 - 1$$



Другой треугольный сигнал, смоделированный с помощью ряда Фурье:

$$ft(x,N,L) := \frac{8}{\pi^2}\cdot\left[\sum_{n=1}^{N}\left[\frac{(-1)^{\frac{n-1}{2}}}{n^2}\cdot\sin\left(\frac{n\cdot\pi\cdot x}{L}\right)\cdot\mathrm{mod}(n,2)\right]\right]$$



Пример 4. Пусть требуется смоделировать произвольный электрический сигнал, вид которого на интервале [0,T] задан кусочно-линейной функцией f(t,T).

Пусть период $T := 1$

Определим кусочно-линейную функцию следующим образом:

$$f(t,T) := \begin{cases} t & \text{if } 0 \le t \le 0.4\cdot T \\ 0.4 & \text{if } 0.4\cdot T \le t \le 0.6\cdot T \\ -t+1 & \text{if } 0.6\cdot T \le t \le T \end{cases}$$

Пусть $w := 2\cdot\frac{\pi}{T}$

Вычислим коэффициенты ряда Фурье:

$$a_0 := \frac{\int_0^T f(t,T)\,dt}{0.5T}$$

$$A(k,f,T) := \frac{\int_0^T f(t,T)\cdot\cos(k\cdot w\cdot t)\,dt}{0.5\cdot T}$$

$$B(k,f,T) := \frac{\int_0^T f(t,T)\cdot\sin(k\cdot w\cdot t)\,dt}{0.5\cdot T}$$

Зададим параметр Nf - количество гармонических членов ряда, сумма которых достаточно хорошо приближается к функции f(t,T). Определим переменную-диапазон (или ранжированную переменную) k и вычислим гармонические члены ряда при различных значениях k:

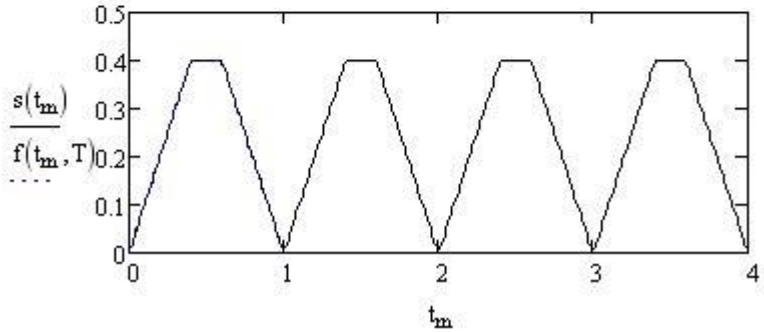
$Nf := 10 \quad k := 1..Nf \qquad a_k := A(k,f,T) \qquad b_k := B(k,f,T)$

Используя ранее вычисленный член ряда a_0 сформируем новую функцию s(t) по формуле ряда Фурье:

$$s(t) := \frac{a_0}{2} + \left[\sum_k \left(a_k\cdot\cos(k\cdot w\cdot t) + b_k\cdot\sin(k\cdot w\cdot t)\right)\right]$$

Построим новую функцию s(t) на интервале [0,4] с шагом 0,01:

$Np := 100 \quad m := 0..4\cdot Np \qquad t_m := \frac{T\cdot m}{Np}$



Видно, что графики практически полностью совпадают.

Пример 5. Рассмотрим еще одну периодическую кусочно-линейную функцию, определенную на интервале [-T, T] период которой равен 2T.

В данном примере для вычисления коэффициентов ряда Фурье используется функция FurCoeff(f,N,T), которая вычисляет значения коэффициентов при помощи программы, причем записывает их в

матрицу, количество столбцов которой равно N+1. Первый (точнее нулевой) столбец заполняется в самом начале, а затем в цикле подсчитываются остальные столбцы. В самом конце функции выполняется операция транспонирования матрицы.

Параметр N определяет количество членов ряда Фурье, достаточное для хорошего приближения к функции f(x,T).

Определение кусочно-линейной периодической функции:

$$f(x,T) := \begin{cases} 1 & \text{if } 0 \le x \le T \\ -x & \text{if } -T \le x < 0 \\ f(x-2,T) & \text{if } x > T \end{cases}$$

Пусть правая граница интервала $T := 1$

Программа для вычисления коэффициентов ряда Фурье:

$$\text{FurCoeff}(f,N,T) := \left| \begin{array}{l} M^{\langle 0 \rangle} \leftarrow \begin{pmatrix} \frac{1}{2\cdot T}\cdot\int_{-T}^{T} f(x,T)\,dx \\ 0 \end{pmatrix} \\ \text{for } n \in 1..N \\ \quad M^{\langle n \rangle} \leftarrow \begin{pmatrix} \frac{1}{T}\cdot\int_{-T}^{T} f(x,T)\cdot\cos\left(\frac{n\cdot\pi\cdot x}{T}\right)dx \\ \frac{1}{T}\cdot\int_{-T}^{T} f(x,T)\cdot\sin\left(\frac{n\cdot\pi\cdot x}{T}\right)dx \end{pmatrix} \\ M^{T} \end{array} \right.$$

Порядок аппроксимации, обеспечивающий хорошее приближение к функции f(x,T):

$Nt := 20$

Вычисляем матрицу коэффициентов Фурье: $res := \text{FurCoeff}(f,Nt,T)$

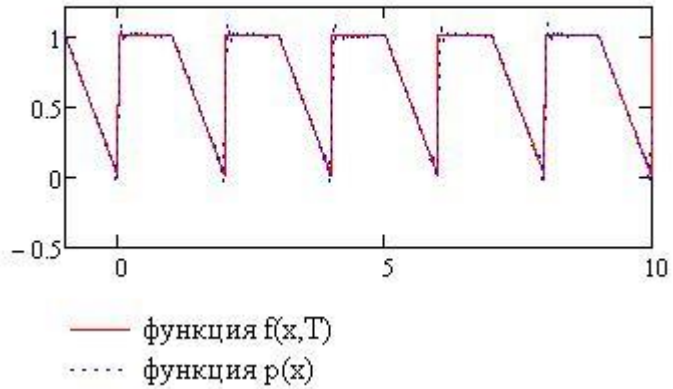
Выделяем вектор-столбцы коэффициентов A и B:

$A := res^{\langle 0 \rangle}$ $B := res^{\langle 1 \rangle}$

Определим функцию p(x) - ряд Фурье:

$$p(x) := A_0 + \sum_{n=1}^{Nt} \left(A_n \cdot \cos\left(\frac{n \cdot \pi \cdot x}{T}\right) + B_n \cdot \sin\left(\frac{n \cdot \pi \cdot x}{T}\right) \right)$$

Построим графики функций p(x) и f(x,T):



Видно, что ряд Фурье при Nt=20 обеспечивает достаточно хорошее приближение к функции f(x,T).

Задания для лабораторных работ

Лабораторная работа 1

Задание на лабораторную работу:

1) Используя математическую модель интегрирующей RC-цепи, решить задачу автоматического подбора параметра. Величина выходного напряжения, снимаемого на конденсаторе к моменту времени t_k после подачи на вход прямоугольного единичного сигнала (рис. 2) должна достигнуть значения $U_{тр}$. Подобрать такое значение сопротивления резистора R при определенной заданной ёмкости конденсатора, которое обеспечило бы решение

поставленного технического задания. Значения конечного времени t_k и ёмкости конденсатора С взять из таблицы по указанию преподавателя. Обратить внимание на табличные единицы измерения и учесть, что в программах следует задавать емкость в фарадах.

Таблица 1. Значения t_k, ёмкости С и требуемого выходного напряжения.

Вариант	t_k	С, мкФ	$U_{тр}$, В
1	0.8	0.8	0.7
2	0.82	0.85	0.71
3	0.84	0.9	0.72
4	0.86	0.95	0.73
5	0.88	1.00	0.74
6	0.9	1.05	0.75
7	0.92	1.1	0.76
8	0.94	1.15	0.77
9	0.96	1.2	0.78
10	0.98	1.25	0.79
11	1.00	1.3	0.8
12	1.02	1.35	0.81

2) Используя математическую модель дифференцирующей RC-цепи, решить задачу автоматического подбора параметра. Величина выходного напряжения, снимаемого на резисторе к моменту времени t_k после подачи на вход прямоугольного единичного сигнала (рис. 2) должна достигнуть значения $U_{тр}$. Подобрать такое значение ёмкости конденсатора С при определенном заданном сопротивлении резистора, которое обеспечило бы решение поставленного технического задания. Значения конечного времени t_k и

сопротивления резистора R взять из таблицы по указанию преподавателя. Обратить внимание на табличные единицы измерения и учесть, что в программах следует задавать сопротивление в омах.

Таблица 2. Значения t_k, сопротивления R и выходного напряжения $U_{тр}$.

Вариант	t_k	R (МОм)	$U_{тр}$ (В)
1	0.2	0.81	0.3
2	0.25	0.84	0.28
3	0.3	0.87	0.26
4	0.35	0.9	0.24
5	0.4	0.93	0.22
6	0.45	0.96	0.2
7	0.5	1.00	0.18
8	0.55	1.13	0.16
9	0.6	1.2	0.14
10	0.65	1.23	0.12
11	0.7	1.27	0.1
12	0.75	1.3	0.08

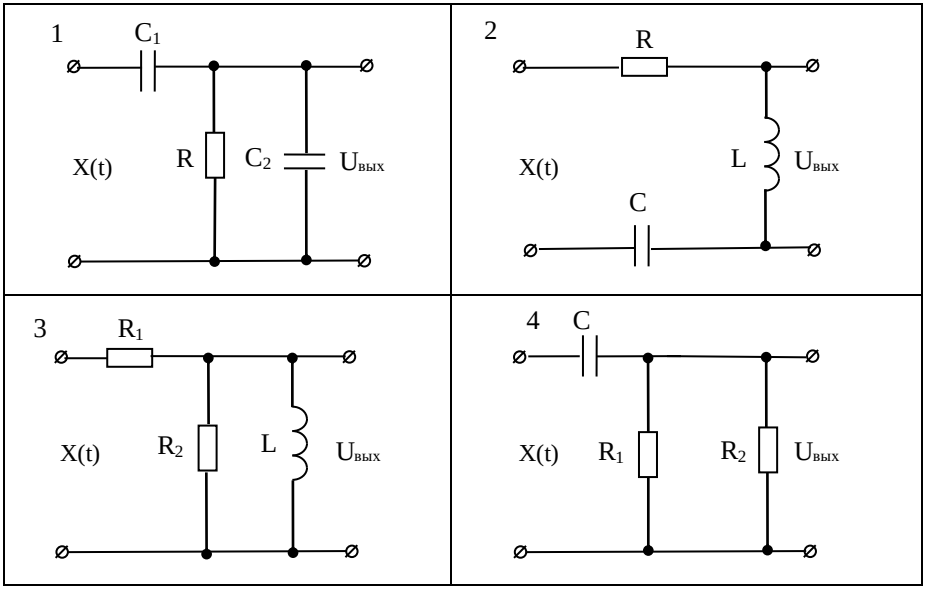
3) В качестве отчета по лабораторной работе представить преподавателю программу на языке Си, строящую графики выходного напряжения и выводящую на экран значения выходного напряжения в конечный момент времени при различных значениях R для первого и C для второго заданий, соответственно. Программа также должна выводить на экран автоматически подобранные значения сопротивления резистора R для первой задачи и ёмкости конденсатора C для второй.

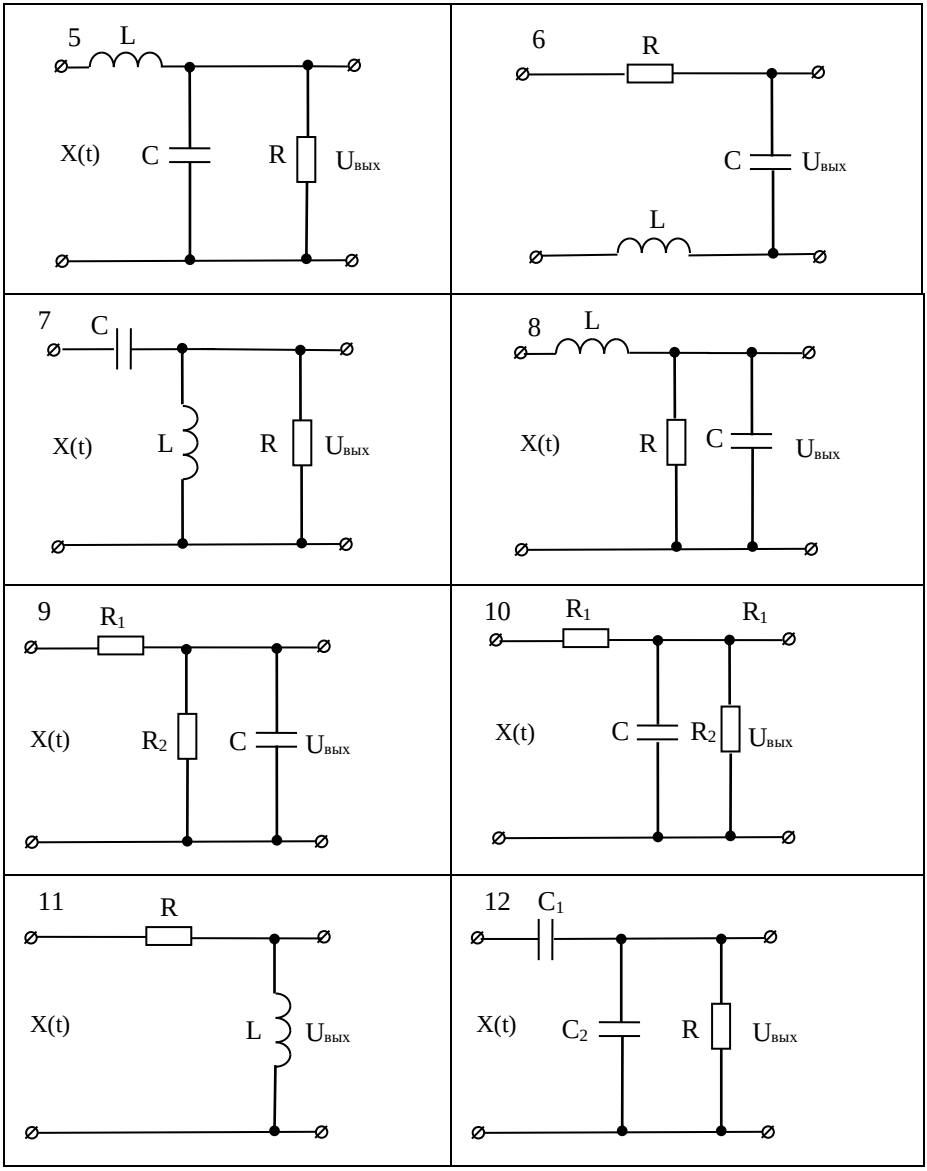
Лабораторная работа 2

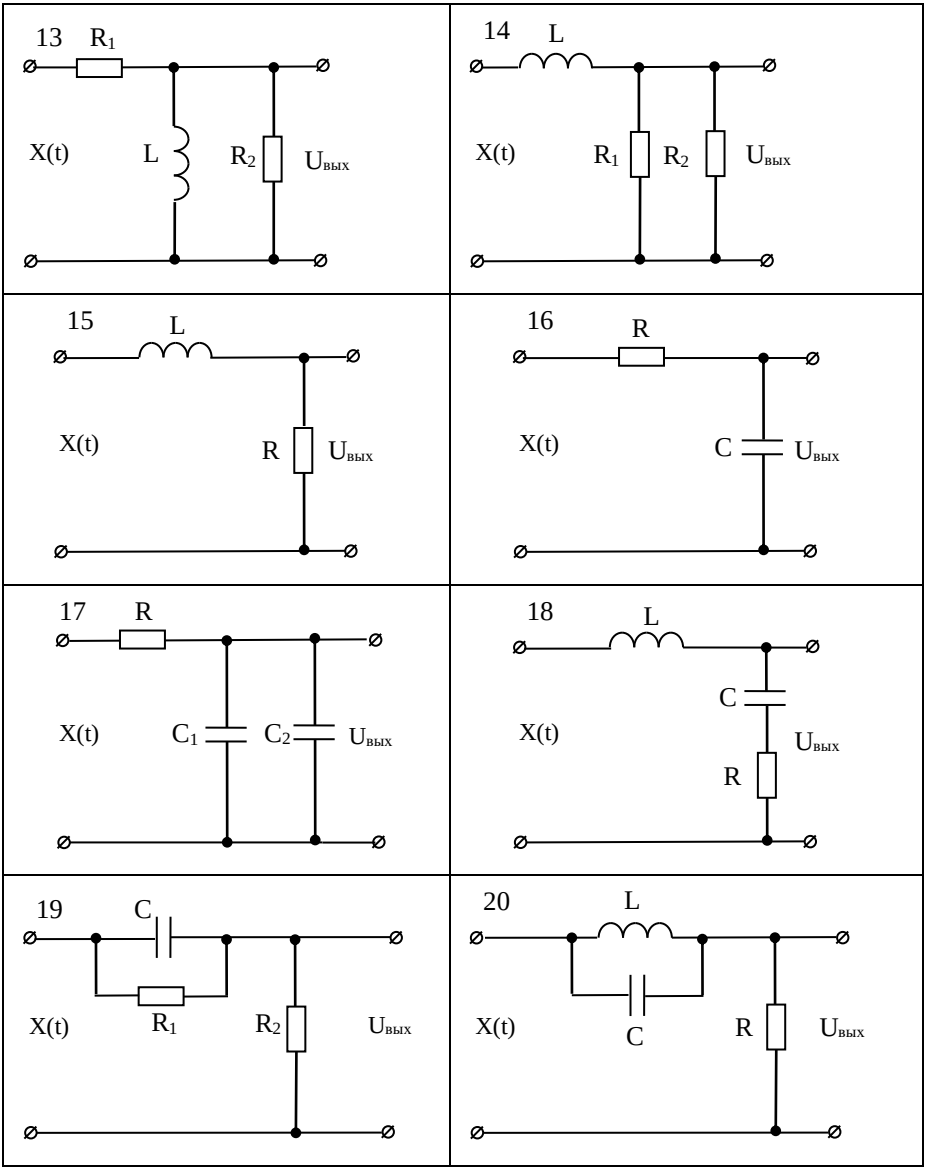
Задание на лабораторную работу:

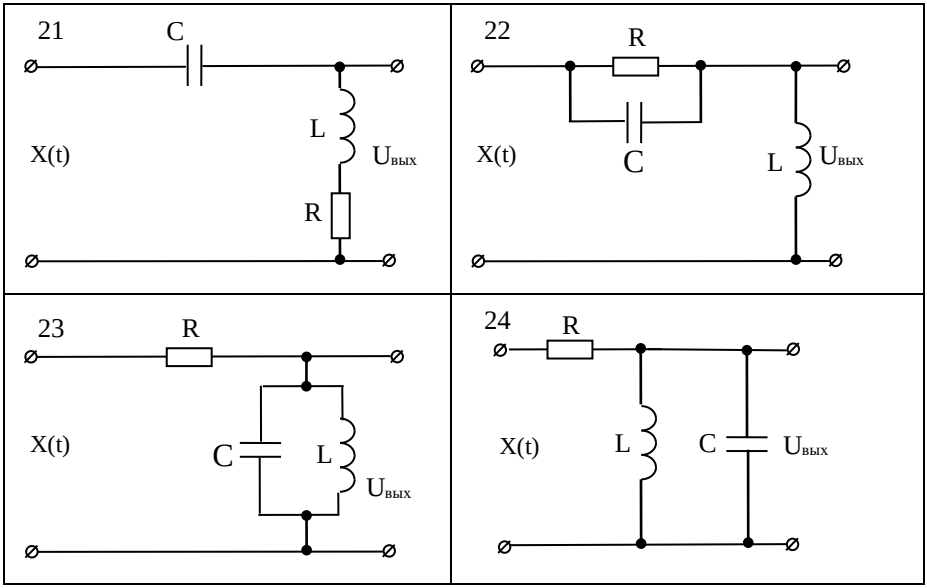
1) составить математическую модель динамической системы в виде дифференциального уравнения с неизвестной функцией $U_{вых}$ по варианту из таблицы 3, выданному преподавателем. Учесть, что X(t) – функция, задающая входное напряжение цепи. Вид входного напряжения также следует уточнить у преподавателя;
2) решить полученное дифференциальное уравнение в пакете MathCAD или в программе на языке Си;
3) исследовать зависимость выходного напряжения $U_{вых}$ от параметров цепи.
4) Представить преподавателю отчет, содержащий вывод уравнения, программу на языке Си или решение, полученное в пакете MathCAD.
5) Защитить работу.

Таблица 3. Варианты электрических схем.




5
L
X(t)
C
R
Uвых
6
R
C
Uвых
L
7
C
X(t)
L
R
Uвых
8
L
X(t)
R
C
Uвых
9
R1
X(t)
R2
C
Uвых
10
R1
R1
X(t)
C
R2
Uвых
11
R
X(t)
L
Uвых
12
C1
X(t)
C2
R
Uвых


13
R1
X(t)
L
R2
Uвых
14
L
X(t)
R1
R2
Uвых
15
L
X(t)
R
Uвых
16
R
X(t)
C
Uвых
17
R
X(t)
C1
C2
Uвых
18
L
C
X(t)
Uвых
R
19
C
X(t)
R1
R2
Uвых
20
L
X(t)
C
R
Uвых

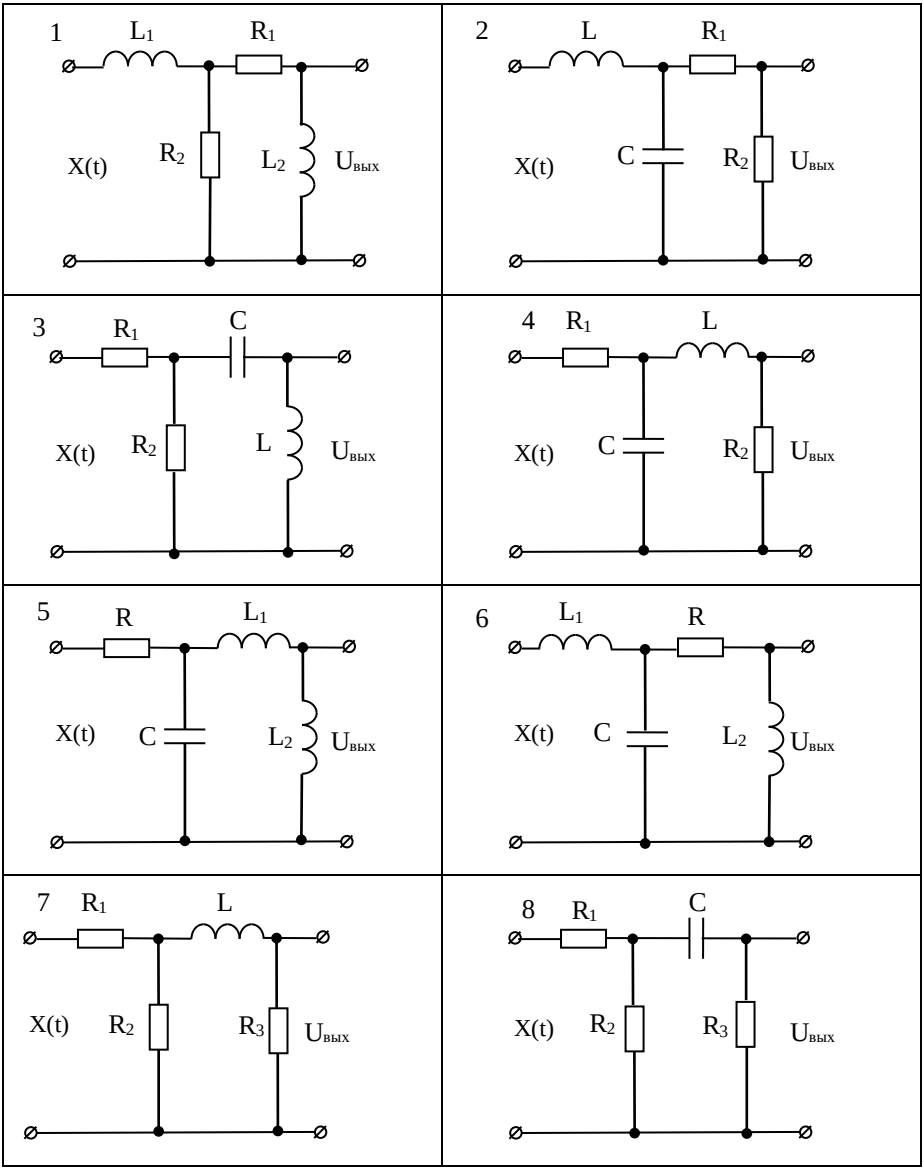


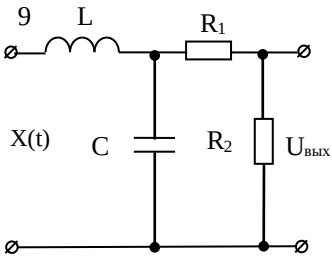
Лабораторная работа 3

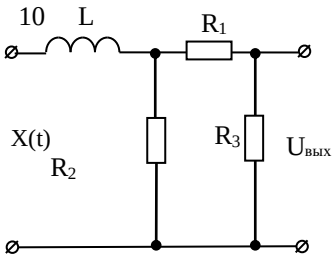
Задание на лабораторную работу:

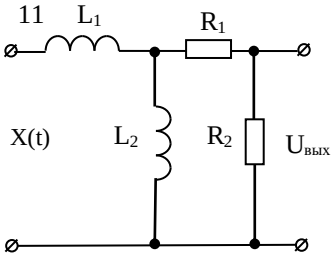
1) составить математическую модель динамической системы в виде дифференциального уравнения с неизвестной функцией $U_{вых}$ по варианту из таблицы 4, выданному преподавателем. Учесть, что X(t) – функция, задающая входное напряжение цепи. Вид входного напряжения также следует уточнить у преподавателя;
2) решить полученное дифференциальное уравнение в пакете MathCAD или в программе на языке Си;
3) исследовать зависимость выходного напряжения $U_{вых}$ от параметров цепи.
4) Представить преподавателю отчет, содержащий вывод уравнения, программу на языке Си или решение, полученное в пакете MathCAD.
5) Защитить работу.

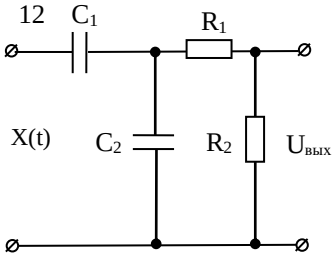
Таблица 4. Варианты электрических схем.

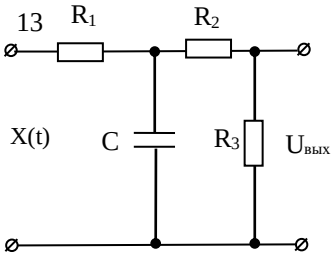


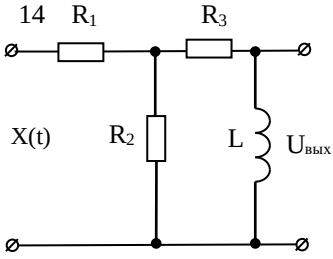

9
L
R1
X(t)
C
R2
Uвых

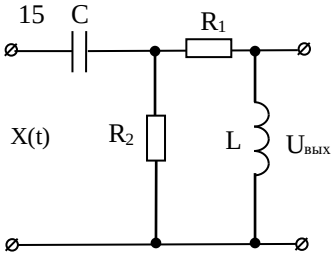

10
L
R1
X(t)
R2
R3
Uвых

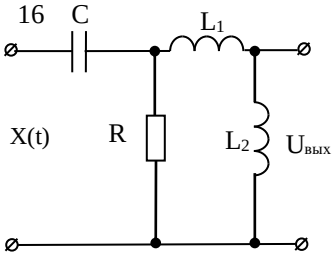

11
L1
R1
X(t)
L2
R2
Uвых

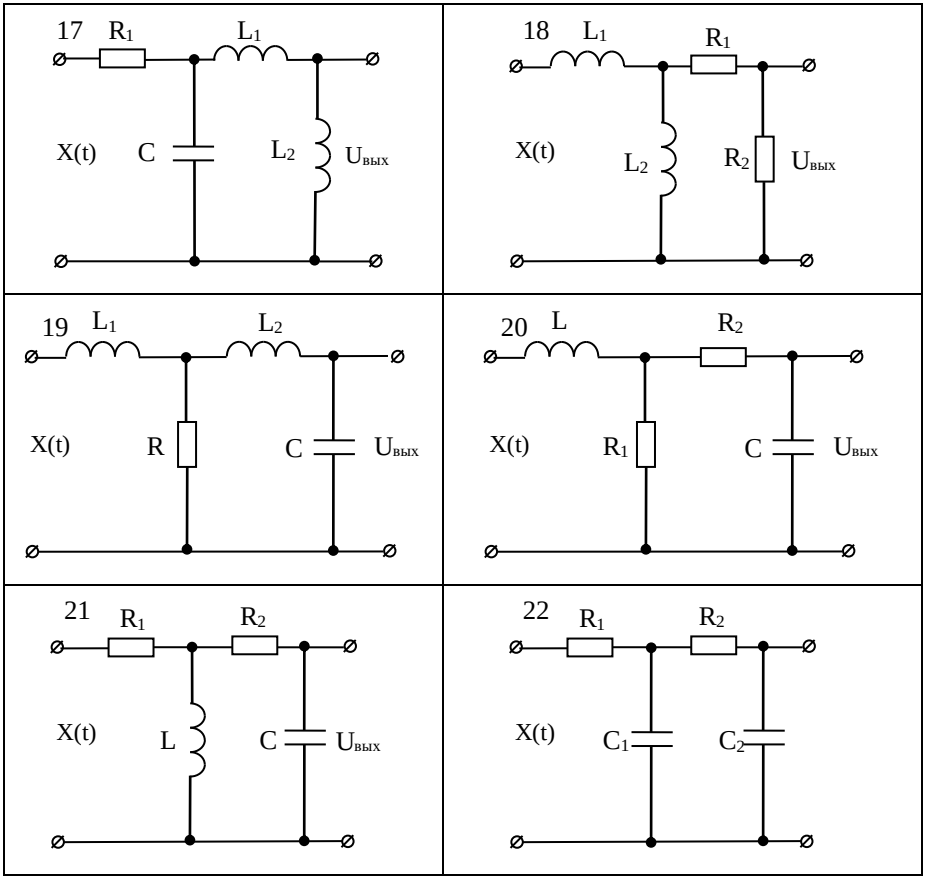

12
C1
R1
X(t)
C2
R2
Uвых


13
R1
R2
X(t)
C
R3
Uвых


14
R1
R3
X(t)
R2
L
Uвых


15
C
R1
X(t)
R2
L
Uвых


16
C
L1
X(t)
R
L2
Uвых



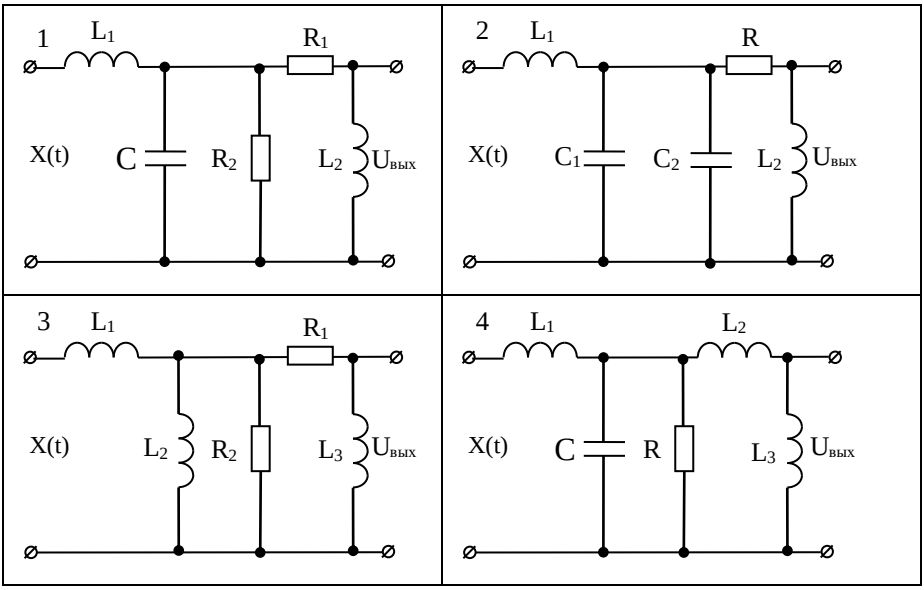
Лабораторная работа 4

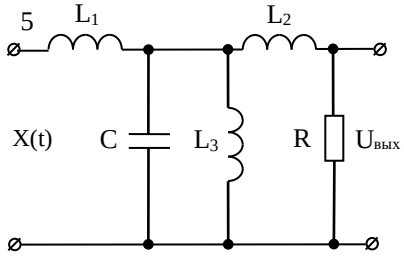
Задание на лабораторную работу:

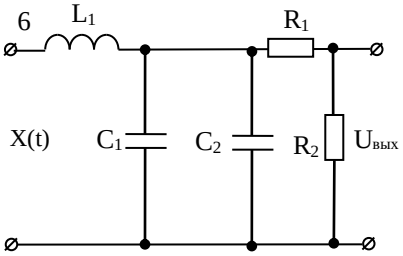
1) составить математическую модель динамической системы в виде дифференциального уравнения с неизвестной функцией $U_{вых}$ по варианту из таблицы 5, выданному преподавателем. Учесть, что X(t) – функция, задающая входное напряжение цепи. Вид входного напряжения также следует уточнить у преподавателя;

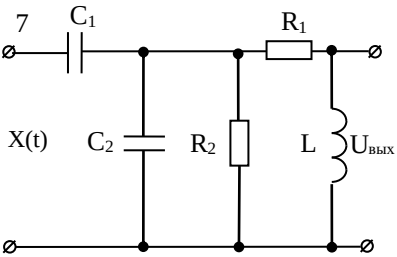
2) решить полученное дифференциальное уравнение в пакете MathCAD или в программе на языке Си;
3) исследовать зависимость выходного напряжения $U_{вых}$ от параметров цепи.
4) Представить преподавателю отчет, содержащий вывод уравнения, программу на языке Си или решение, полученное в пакете MathCAD.
5) Защитить работу.

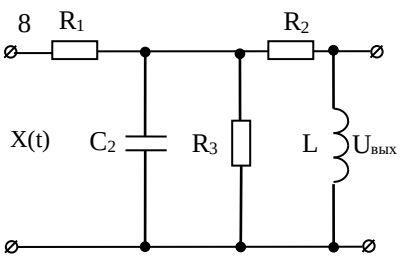
Таблица 5. Варианты электрических схем.

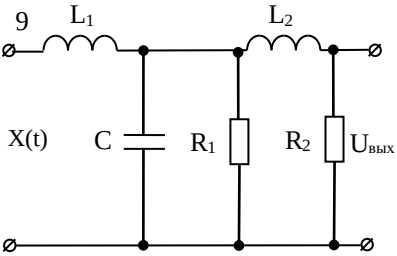


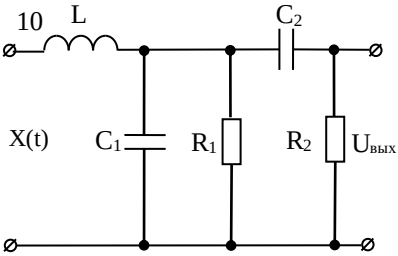


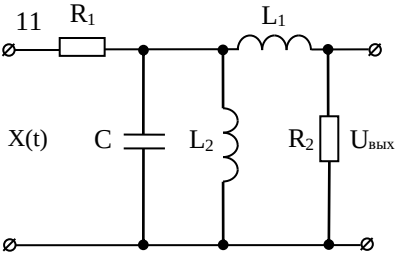


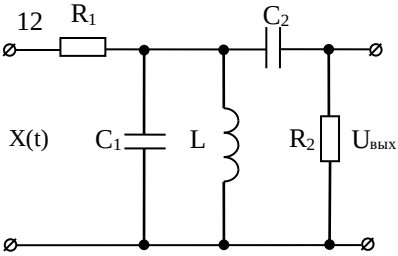

7
C1
R1
X(t)
C2
R2
L
Uвых


8
R1
R2
X(t)
C2
R3
L
Uвых


9
L1
L2
X(t)
C
R1
R2
Uвых


10
L
C2
X(t)
C1
R1
R2
Uвых


11
R1
L1
X(t)
C
L2
R2
Uвых


12
R1
C2
X(t)
C1
L
R2
Uвых

Оглавление

Введение 3
1.Компьютерное моделирование 4
2.Получение математических моделей электротехнических цепей 6
2.1. Математические модели неразветвленных цепей 6
2.1.1. Математическая модель RC-цепи. Пример 1 6
2.1.2. Математическая модель RC-цепи. Пример 2 8
2.1.3. Математическая модель RC-цепи. Пример 3 9
2.2.1. RLC-цепь (колебательный контур). Пример 1 10
2.2.2. RLC-цепь. Пример 2 12
2.2.3. RLC-цепь. Построение математической модели с использованием преобразования Лапласа.....13
2.3. Задача автоматического подбора параметров цепи ...14
2.4. Математические модели разветвленных цепей.....21
2.4.1. Математическая модель разветвлённой RC-цепи. ..21
2.5. Математические модели электрических цепей с нелинейностями. Однополупериодный выпрямитель 22
2.6. Моделирование электрических цепей в пакете MathCAD 14.....25
2.7. Моделирование электрических сигналов в пакете MathCAD14 30
Задания для лабораторных работ 36
Лабораторная работа 1 36
Лабораторная работа 2 39
Лабораторная работа 3 42
Лабораторная работа 4 45

Итоговый тест по дисциплине «Информатика»

специальность 26.05.07 «Эксплуатация судового электрооборудования и средств автоматики»
Профиль (специализация) «Эксплуатация судового электрооборудования и средств автоматики»

ОПК-5 Способен использовать современные информационные технологии и программные средства при решении задач профессиональной деятельности, обеспечивая выполнение требований информационной безопасности

№ п/п	Виды тестовых заданий	Тестовое задание
1	Тестовое задание закрытого типа с выбором ответа	*Прочитайте текст и выберите правильный ответ.* Какой компонент процессора отвечает за выполнение арифметических и логических операций? 1. регистр; 2. АЛУ (арифметико-логическое устройство); 3. УУ (устройство управления); 4. кэш-память. *Запишите цифру правильного ответа*
2	Тестовое задание закрытого типа с выбором ответа	*Прочитайте текст и выберите правильный ответ.* Какой протокол используется для передачи веб-страниц в Интернете 1. FTP; 2. SMTP; 3. HTTP; 4. TCP. *Запишите цифру правильного ответа*
3	Тестовое задание закрытого типа с выбором ответа	*Прочитайте текст и выберите правильный ответ.* Что такое облачные технологии? 1. технологии обработки данных на удалённых серверах; 2. порядок использования локальных серверов организации; 3. технологии резервного копирования на жёсткие диски; 4. использование программ для работы с электронной почтой. *Запишите цифру правильного ответа*
4	Тестовое задание закрытого типа с выбором ответа	*Прочитайте текст и выберите правильный ответ.* Какой структуре данных соответствует принцип «последний вошёл – первый вышел» (LIFO)? 1. очередь; 2. стек; 3. список; 4. дерево. *Запишите цифру правильного ответа*
5	Тестовое задание закрытого типа с выбором ответа	*Прочитайте текст и выберите правильный ответ.* Что такое MAC-адрес? 1. числовой идентификатор сети; 2. уникальный физический адрес сетевого устройства;

<table>
<tr><td></td><td></td><td>3. доменное имя устройства;
4. IP-адрес устройства.
Запишите цифру правильного ответа</td></tr>
<tr><td>6</td><td>Тестовое задание закрытого типа на установление соответствия</td><td>Прочитайте текст и установите соответствие.
Соотнесите сетевой протокол с его функцией:
А) HTTP;
Б) FTP;
В) SMTP;
Г) DNS.

1. Передача файлов между узлами сети;
2. Преобразование доменных имён в IP-адреса;
3. Передача гипертекста (веб-страниц);
4. Отправка электронной почты.
Запишите выбранные цифры под соответствующими буквами:
<table>
<tr><td>А</td><td>Б</td><td>В</td><td>Г</td></tr>
<tr><td></td><td></td><td></td><td></td></tr>
</table></td></tr>
<tr><td>7</td><td>Тестовое задание закрытого типа на установление соответствия</td><td>Прочитайте текст и установите соответствие.
Установите соответствие между видом кибератаки и её описанием:
А) Фишинг;
Б) DDoS;
В) Вредоносное ПО;
Г) Социальная инженерия.

1. Массовая отправка запросов для перегрузки сервера;
2. Использование обмана для получения конфиденциальных данных;
3. Рассылка поддельных писем и использование поддельных сайтов для кражи логинов и паролей;
4. Программы, наносящие вред системе (вирусы, трояны).
Запишите выбранные цифры под соответствующими буквами:
<table>
<tr><td>А</td><td>Б</td><td>В</td><td>Г</td></tr>
<tr><td></td><td></td><td></td><td></td></tr>
</table></td></tr>
<tr><td>8</td><td>Комбинированное задание (с выбором нескольких ответов и обоснованием выбора)</td><td>Прочитайте текст, выберите правильный ответ(ы) и запишите аргументы, обосновывающие выбор ответа(ов).
Какие свойства должны быть у надёжного пароля?
1. длина не менее 8 символов;
2. использование только букв;
3. наличие цифр и специальных символов;
4. предсказуемость (например, дата рождения);
5. уникальность (не повторяется на других сервисах).
Запишите цифру(цифры) правильного ответа(ов) и обоснуйте каждый из них.</td></tr>
</table>

9	Открытое задание (с развернутым ответом)	*Прочитайте текст и запишите развернутый обоснованный ответ.* На рабочем листе табличного процессора Excel в ячейку A1 введено число 10. В ячейку B1 введено число 3. В ячейку C1 введена формула =A$1 – $B1 Какой результат получится в ячейке D2 при копировании в нее формулы из ячейки C1? *Обоснуйте ответ.*
10	Открытое задание (с развернутым ответом)	*Прочитайте текст и запишите развернутый обоснованный ответ.* *Фрагмент программы на языке C++:* *S=0;* *for (i=1; i<4; i++)* *{* *S+=2*i;* *}* *Вычислите значение переменной S,обоснуйте ответ,выписывая последовательно каждое новое значение S на каждом шаге цикла.*

ПК-6 Способен осуществлять безопасное техническое использование, техническое обслуживание судовой компьютерной информационной системы в соответствии с международными и национальными требованиями

№ п/п	Виды тестовых заданий	Тестовое задание
1	Тестовое задание закрытого типа с выбором ответа	*Прочитайте текст и выберите правильный ответ.* Что обязательно следует сделать перед обновлением программного обеспечения на судовых компьютерах? 1) уведомить капитана; 2) создать резервную копию данных и конфигураций; 3) отключить систему от сети; 4) провести антивирусную проверку *Запишите цифру правильного ответа.*
2	Тестовое задание закрытого типа с выбором ответа	*Прочитайте текст и выберите правильный ответ.* Что является признаком перегрева серверного оборудования? 1) тихий шум вентиляторов; 2) повышение температуры корпуса, аварийные сигналы; 3) мигание зелёных индикаторов; 4) увеличение скорости сети *Запишите цифру правильного ответа.*
3	Тестовое задание закрытого типа с выбором ответа	*Прочитайте текст и выберите правильный ответ.* Как следует хранить запасные компоненты (платы, диски и т. п.) на судне? 1) в открытом ящике; 2) в антистатической упаковке, в сухом и защищённом от вибраций месте; 3) рядом с источниками тепла; 4) в каюте ответственного лица *Запишите цифру правильного ответа.*
4	Тестовое задание закрытого типа с выбором ответа	*Прочитайте текст и выберите правильный ответ.* Что такое IP-адрес версии IPv4? 1) шестнадцатеричный код длиной 2 байта; 2) последовательность из 4 байт, записанная в десятичном виде через точки;

<table>
<tr><td></td><td></td><td>3) MAC-адрес сетевого устройства;
4) доменное имя сайта.
Запишите цифру правильного ответа.</td></tr>
<tr><td>5</td><td>Тестовое задание закрытого типа с выбором ответа</td><td>Прочитайте текст и выберите правильный ответ.
Какой документ фиксирует все проведённые технические работы с ИТ-системой?
1) судовой журнал;
2) бортовой компьютер;
3) журнал технического обслуживания;
4) личная записка механика
Запишите цифру правильного ответа.</td></tr>
<tr><td>6</td><td>Тестовое задание закрытого типа на установление соответствия</td><td>Прочитайте текст и установите соответствие.
Установите соответствие между этапами жизненного цикла информационной системы и действиями:
А) ввод в эксплуатацию;
Б) эксплуатация;
В) модернизация;
Г) списание.

1) Замена компонентов, расширение функционала, переход на новые стандарты;
2) Настройка, приёмо-сдаточные испытания;
3) Регулярное ТО, мониторинг, обновление ПО;
4) Архивация данных, демонтаж, утилизация с соблюдением экологических норм.
Запишите выбранные цифры под соответствующими буквами:
<table>
<tr><td>А</td><td>Б</td><td>В</td><td>Г</td></tr>
<tr><td></td><td></td><td></td><td></td></tr>
</table>
</td></tr>
<tr><td>7</td><td>Тестовое задание закрытого типа на установление соответствия</td><td>Прочитайте текст и установите соответствие.
Установите соответствие между компонентами судовой информационной системы и их функциями:
А) АИС (Automatic Identification System);
Б) ЭНК (электронная навигационная карта);
В) СУДС (система управления движением судов);
Г) ГНСС (глобальная навигационная спутниковая система).

1) Предоставление координат, скорости и времени с высокой точностью;
2) Мониторинг и управление судопотоком в портах и узкостях с использованием данных АИС и радаров;
3) Отображение картографической информации и прокладка маршрута;
4) Обмен статической, динамической и рейсовой информацией между судами и береговыми службами.
Запишите выбранные цифры под соответствующими буквами:
<table>
<tr><td>А</td><td>Б</td><td>В</td><td>Г</td></tr>
<tr><td></td><td></td><td></td><td></td></tr>
</table>
</td></tr>
<tr><td>8</td><td>Тестовое задание закрытого типа на установление последовательности</td><td>Прочитайте текст и установите правильную последовательность действий.

Алгоритм реагирования на сбой информационной системы в рейсе:
1) уведомить капитана и береговую службу;
2) зафиксировать время, симптомы и предварительные</td></tr>
</table>

		причины сбоя; 3) перейти на резервные средства (если предусмотрено); 4) приступить к диагностике и устранению неисправности; 5) внести запись в журнал технического обслуживания. *Запишите соответствующую правильную последовательность цифр слева направо:* | | | | | |
9	Открытое задание (с развернутым ответом)	*Прочитайте текст и запишите развернутый обоснованный ответ.* Как правильно утилизировать носители информации (жёсткие диски, флеш-накопители) с конфиденциальными данными? Почему недопустимо выбросить в мусор или сдать в металлолом? *Обоснуйте ответ.*
10	Открытое задание (с развернутым ответом)	*Прочитайте текст и запишите развернутый обоснованный ответ.* Как обеспечить непрерывность работы судовой информационной системы при отключении основного электропитания? *Обоснуйте ответ.*