

«

»

“ ”

“ ” . . . . .

# РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ Информатика

: 24.03.03

, :

: 1, : 1

		<b>1</b>
<b>1</b>	( )	4
<b>2</b>		144
<b>3</b>	, .	87
<b>4</b>	, .	18
<b>5</b>	, .	0
<b>6</b>	, .	54
<b>7</b>	, .	18
<b>8</b>	, .	2
<b>9</b>	, .	13
<b>10</b>	, .	57
<b>11</b>	( , , , )	
<b>12</b>		

( ): 24.03.03

1413 03.12.2015 ., : 31.12.2015 .

: 1,

( ): 24.03.03

, 5/1 20.06.2017  
, 6 20.06.2017

, 5 21.06.2017

:

, . . . . . . . . .

:

, . . . . . . . .  
, . . . . . . . .

:

. . . .

# 1.

1.1

<b>Компетенция ФГОС: ОК.8 осознанием сущности и значения информации в развитии современного общества, владение основными методами, способами и средствами получения, хранения, переработки информации; в части следующих результатов обучения:</b>	
1.	
2.	, ,
1.	
2.	
3.	
<b>Компетенция ФГОС: ОК.9 наличием навыков работы с компьютером как со средством управления информацией; в части следующих результатов обучения:</b>	
1.	
2.	
3.	
4.	
5.	, ,
<b>Компетенция ФГОС: ОПК.2 умением получать, собирать, систематизировать и анализировать информацию в области профессиональной деятельности; в части следующих результатов обучения:</b>	
1.	- ,

# 2.

2.1

, , , ) (	
<b>.2. 1</b> - ,	
1.уметь проводить библиографическую и информационно-поисковую работы, использовать ее результаты при решении профессиональных задач и оформлении научных трудов	;
<b>.8. 1</b>	
2.знать правовые основы информационной безопасности и принципы защиты авторского права на программные продукты	;
<b>.8. 2</b> ,	
3.знать сущность и значение информации в развитии современного общества, опасности и угроз, возникающие в этом процессе	;
<b>.8. 1</b>	
4.уметь использовать языки и системы программирования для решения профессиональных задач	; ;
<b>.8. 2</b>	

5.уметь осуществлять поиск информации в локальных и глобальных сетях		
<b>.8. 3</b>		
6.уметь оценивать состояние и тенденции развития информационных технологий и информатики в современном обществе		
<b>.9. 1</b>		
7.владеть персональным компьютером как средством управления информацией		
<b>.9. 2</b>		
8.уметь использовать специализированные программные средства при решении профессиональных задач		
<b>.9. 3</b>		
9.уметь использовать элементарные навыки алгоритмизации и программирования на одном из языков высокого уровня как средство программного моделирования изучаемых объектов и процессов		
<b>.9. 4</b>		
10.уметь пользоваться наиболее распространенными офисными и математическими пакетами прикладных программ		
<b>.9. 5</b>		
11.уметь применять основные методы, способы и средства получения, хранения и переработки информации с помощью компьютеров и компьютерных средств		

### 3.

#### 3.1

<b>: 1</b>			
1.	0	2	10, 2, 3
2.	0	2	2, 3, 6
3.	0	2	4, 8, 9
4.	0	2	11, 8, 9
5.	0	2	4, 9

:			
6.	.	0	3 4, 8, 9
:			
7.	.	0	3 4, 7, 9
8.		0	2 1, 11, 5, 7

3.2

	,	.		
: 1				
, , . , ,				
1.	3	12	10, 4, 7, 8, 9	(Excel, MathCAD, FORTRAN).
:				
2.	3	6	10, 4, 7, 8, 9	(Excel, MathCAD, FORTRAN).
:				
3.	3	8	10, 11, 4, 7, 9	(Excel, MathCAD, FORTRAN).
:				
4.	3	8	10, 11, 4, 7, 8, 9	(Excel, MathCAD, FORTRAN).
:				

5.	3	10	10, 11, 4, 7, 8, 9	(Excel, MathCAD, FORTRAN).
5.	3	10	10, 11, 4, 8, 9	(Excel, MathCAD, FORTRAN).

#### 4.

: 1				
1		1, 10, 11, 4, 5, 6, 7, 8, 9	40	13
: " " 1 / . . . - ;[ .: . . .]. - , 2003. - 59 .: .. - : <a href="http://www.library.nstu.ru/fulltext/metodics/2003/2558.rar">http://www.library.nstu.ru/fulltext/metodics/2003/2558.rar</a>				
2		1, 10, 11, 4, 5, 6, 7, 8, 9	9	0
: " " 1 / . . . - ;[ .: . . .]. - , 2003. - 59 .: .. - : <a href="http://www.library.nstu.ru/fulltext/metodics/2003/2558.rar">http://www.library.nstu.ru/fulltext/metodics/2003/2558.rar</a>				
3		1, 10, 11, 2, 3, 4, 5, 7, 8, 9	8	0
: " " 1 / . . . - ;[ .: . . .]. - , 2003. - 59 .: .. - : <a href="http://www.library.nstu.ru/fulltext/metodics/2003/2558.rar">http://www.library.nstu.ru/fulltext/metodics/2003/2558.rar</a> [ . . . ]: - / . . . , . . . ; . . . - . - , [2011]. - : <a href="http://courses.edu.nstu.ru/index.php?show=155&amp;curs=692">http://courses.edu.nstu.ru/index.php?show=155&amp;curs=692</a> . - . . : 280200 - / . . . - ;[ . . . ]. - , 2007. - 46, [1] .: ., , .. - : <a href="http://www.library.nstu.ru/fulltext/metodics/2007/07_Korotaeva.rar">http://www.library.nstu.ru/fulltext/metodics/2007/07_Korotaeva.rar</a> : 1-2 / . . . - ;[ .: . . .] , . ]. - , 2010. - 43, [1] .: ., .. - : <a href="http://www.library.nstu.ru/fulltext/metodics/2010/2010_3917.pdf">http://www.library.nstu.ru/fulltext/metodics/2010/2010_3917.pdf</a>				

#### 5.

,  
- ( . 5.1).

5.1

	-
	e-mail:krasnorutskiy@corp.nstu.ru; :vk.com

	e-mail;
	e-mail;
	;

6.

( ),

-  
15-

ECTS.

. 6.1.

6.1

: 1	
РГЗ:	80
Зачет:	20

6.2

6.2

.8	1.		+
	2. ,		+
	1.		+
	2.		+
	3.		+
.9	1.		+
	2.	+	+
	3.	+	+
	4.	+	+
	5. , ,		+

.2	1.	-	,		+
----	----	---	---	--	---

1

## 7.

1. Информатика. Базовый курс : [учебное пособие для втузов] / под ред. С. В. Симоновича. - СПб. [и др.], 2008. - 639 с. : ил.. - На тит. л.: Издательская программа 300 лучших учебников для высшей школы.
2. Корнеев И. К. Информационные технологии : учебник / И. К. Корнеев, Г. Н. Ксандопуло, В. А. Машурцев ; Гос. ун-т упр. - Москва, 2007. - 222 с. : ил.
3. Информатика [Электронный ресурс] : учебник / Новосиб. гос. техн. ун-т ; [О. К. Альсова и др.]. - Новосибирск, 2012. - 1 электрон. опт. диск (CD-ROM). - Режим доступа: [http://elibrary.nstu.ru/source?bib\\_id=vtls000175426](http://elibrary.nstu.ru/source?bib_id=vtls000175426). - Загл. с этикетки диска.
4. Бартеньев О. В. Современный Фортран / О. В. Бартеньев. - М., 2000. - 448 с.
5. Краинский И. Word 2007 : популярный самоучитель / И. Краинский. - СПб. [и др.], 2008. - 240 с. : ил.
6. Атапин В. Г. Сопротивление материалов : учебник / В. Г. Атапин, А. Н. Пель, А. И. Темников. - Новосибирск, 2006. - 555 с. : ил.. - Режим доступа: <http://www.ciu.nstu.ru/fulltext/textbooks/2006/atapin.pdf>

1. Плис А. И. Mathcad. Математический практикум для инженеров и экономистов : учебное пособие для вузов / А. И. Плис, Н. А. Сливина. - М., 2003. - 655 с. : ил.
2. Плис А. И. Лабораторный практикум по высшей математике : учебное пособие для втузов / А. И. Плис, Н. А. Сливина. - М., 1983. - 208 с.
3. Бартеньев О. В. Современный Фортран : [учебное пособие] / О. В. Бартеньев. - М., 1998. - 397 с. : табл.

1. ЭБС НГТУ : <http://elibrary.nstu.ru/>
2. ЭБС «Издательство Лань» : <https://e.lanbook.com/>
3. ЭБС IPRbooks : <http://www.iprbookshop.ru/>
4. ЭБС "Znaniy.com" : <http://znaniy.com/>
5. :

## 8.

### 8.1

1. Информатика и программирование : методические указания для проведения лабораторных работ по курсу "Информатика" для 1 курса ФЛА / Новосиб. гос. техн. ун-т ; [сост.: М. А. Леган и др.]. - Новосибирск, 2003. - 59 с. : ил.. - Режим доступа: <http://www.library.nstu.ru/fulltext/metodics/2003/2558.rar>
2. Информатика : лабораторные работы для ФЛА направления 280200 - Защита окружающей среды / Новосиб. гос. техн. ун-т ; [сост. Т. А. Коротаева]. - Новосибирск, 2007. - 46, [1] с. : ил., схемы, табл.. - Режим доступа: [http://www.library.nstu.ru/fulltext/metodics/2007/07\\_Korotaeva.rar](http://www.library.nstu.ru/fulltext/metodics/2007/07_Korotaeva.rar)



3. Компьютерное моделирование работы системы термостабилизации человека в условиях низких температур : методические указания к лабораторным работам, курсовому и дипломному проектированию специальности 160202 и магистерской программы подготовки 160100 для 4-6 курсов ФЛА дневного отделения / Новосиб. гос. техн. ун-т ; [сост.: А. В. Чичиндаев, И. В. Хромова]. - Новосибирск, 2008. - 50, [1] с. : ил. - Режим доступа: [http://elibrary.nstu.ru/source?bib\\_id=vtls000085031](http://elibrary.nstu.ru/source?bib_id=vtls000085031)
4. Компьютерное моделирование теплофизических процессов : методические указания к лабораторным работам для 1-2 курсов специальностей 131100, 070200 дневного и заочного отделения ФЛА / Новосиб. гос. техн. ун-т ; [сост. А. В. Чичиндаев]. - Новосибирск, 2003. - 51 с. : табл. - Режим доступа: <http://www.library.nstu.ru/fulltext/metodics/2003/2546.rar>
5. Численные методы и программирование на Фортране : методические указания для проведения лабораторных работ для 1-2 курсов факультета летательных аппаратов / Новосиб. гос. техн. ун-т ; [сост.: Н. В. Третьякова, П. Е. Рябчиков]. - Новосибирск, 2010. - 43, [1] с. : ил., табл. - Режим доступа: [http://www.library.nstu.ru/fulltext/metodics/2010/2010\\_3917.pdf](http://www.library.nstu.ru/fulltext/metodics/2010/2010_3917.pdf)
6. Сопротивление материалов : методические указания и варианты исходных данных к расчетно-графическим заданиям для 2 курса ФЛА дневной формы обучения (специальности 280200 - Защита окружающей среды, 080502 - Экономика и управление на авиапредприятиях / Новосиб. гос. техн. ун-т ; [сост.: М. А. Леган, А. И. Темников, А. Н. Пель]. - Новосибирск, 2006. - 28, [1] с. - Режим доступа: <http://www.library.nstu.ru/fulltext/metodics/2006/3189.rar>
7. Виноградов А. В. Информатика [Электронный ресурс] : учебно-методическое пособие / А. В. Виноградов, Ю. О. Поляков ; Новосиб. гос. техн. ун-т. - Новосибирск, [2011]. - Режим доступа: <http://courses.edu.nstu.ru/index.php?show=155&curs=692>. - Загл. с экрана.
8. Лабораторные работы по информатике № 1-8 для 2 курса ФЛА (специальность 071300) дневного отделения / Новосиб. гос. техн. ун-т ; [сост.: С. Д. Саленко, К. А. Шатилов]. - Новосибирск, 2001. - 22 с. : ил. - Режим доступа: <http://www.library.nstu.ru/fulltext/metodics/2001/salenko.zip>

## 8.2

### 1 MathCAD

## 9. -

1	( - , , )	

1	( Internet )	

Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Новосибирский государственный технический университет»

Кафедра аэрогидродинамики  
Кафедра прочности летательных аппаратов

“УТВЕРЖДАЮ”  
ДЕКАН ФЛА  
д.т.н., профессор С.Д. Саленко  
“    ”    \_\_\_\_\_ Г.

## ФОНД ОЦЕНОЧНЫХ СРЕДСТВ УЧЕБНОЙ ДИСЦИПЛИНЫ

### Информатика

Образовательная программа: 24.03.03 Баллистика и гидроаэродинамика, профиль:  
Гидроаэродинамика

# 1. Обобщенная структура фонда оценочных средств учебной дисциплины

Обобщенная структура фонда оценочных средств по дисциплине Информатика приведена в Таблице.

Таблица

Формируемые компетенции	Показатели сформированности компетенций (знания, умения, навыки)	Темы	Этапы оценки компетенций	
			Мероприятия текущего контроля (курсовой проект, РГЗ и др.)	Промежуточная аттестация (экзамен, зачет)
ОК.8 осознанием сущности и значения информации в развитии современного общества, владение основными методами, способами и средствами получения, хранения, переработки информации	з1. знать правовые основы информационной безопасности и принципы защиты авторского права на программные продукты	Общая характеристика процессов обработки информации. Представление табличной информации в виде графиков Состав и назначение основных элементов персонального компьютера. Логические основы решения задач		Зачет, вопросы 1-50
ОК.8	з2. знать сущность и значение информации в развитии современного общества, опасности и угроз, возникающие в этом процессе	Общая характеристика процессов обработки информации. Представление табличной информации в виде графиков Состав и назначение основных элементов персонального компьютера. Логические основы решения задач		Зачет, вопросы 1-50
ОК.8	у1. уметь использовать языки и системы программирования для решения профессиональных задач	Алгоритмизация простейших численных методов Вычисление конечных сумм Использование массивов Построение графиков функций Программирование ветвящихся вычислительных процессов Программные единицы. Решение нелинейных уравнений Решение нелинейных уравнений Циклические вычислительные процессы Численное интегрирование Языки программирования. Численное интегрирование		Зачет, вопросы 1-50
ОК.8	у2. уметь осуществлять поиск информации в локальных и глобальных сетях	Основные особенности использования текстовых процессоров при оформлении отчетов		Зачет, вопросы 1-50
ОК.8	у3. уметь оценивать состояние и тенденции развития информационных технологий и информатики в	Состав и назначение основных элементов персонального компьютера. Логические основы решения задач		Зачет, вопросы 1-50

	современном обществе			
ОК.9 наличием навыков работы с компьютером как со средством управления информацией	у1. владеть персональным компьютером как средством управления информацией	Основные особенности использование текстовых процессоров при оформлении отчетов Языки программирования. Численное интегрирование		Зачет, вопросы 1-50
ОК.9	у2. уметь использовать специализированные программные средства при решении профессиональных задач	Использование массивов Построение графиков функций Программирование ветвящихся вычислительных процессов Программные единицы. Решение нелинейных уравнений Решение нелинейных уравнений Типовые алгоритмы обработки массивов Циклические вычислительные процессы Численное интегрирование	РГЗ, разделы 1-2	Зачет, вопросы 1-50
ОК.9	у3. уметь использовать элементарные навыки алгоритмизации и программирования на одном из языков высокого уровня как средство программного моделирования изучаемых объектов и процессов	Алгоритмизация простейших численных методов Вычисление конечных сумм Использование массивов Построение графиков функций Программирование ветвящихся вычислительных процессов Программные единицы. Решение нелинейных уравнений Решение нелинейных уравнений Типовые алгоритмы обработки массивов Циклические вычислительные процессы Численное интегрирование Языки программирования. Численное интегрирование	РГЗ, разделы 1-2	Зачет, вопросы 1-50
ОК.9	у4. уметь пользоваться наиболее распространенными офисными и математическими пакетами прикладных программ	Вычисление конечных сумм Использование массивов Общая характеристика процессов обработки информации. Представление табличной информации в виде графиков Построение графиков функций Программирование ветвящихся вычислительных процессов Решение нелинейных уравнений Численное интегрирование	РГЗ, разделы 1-2	Зачет, вопросы 1-50
ОК.9	у5. уметь применять основные методы, способы и средства получения, хранения и переработки информации с помощью компьютеров и компьютерных средств	Основные особенности использование текстовых процессоров при оформлении отчетов Типовые алгоритмы обработки массивов		Зачет, вопросы 1-50

ОПК.2 умением получать, собирать, систематизировать и анализировать информацию в области профессиональной деятельности	у1. уметь проводить библиографическую и информационно-поисковую работы, использовать ее результаты при решении профессиональных задач и оформлении научных трудов	Основные особенности использование текстовых процессоров при оформлении отчетов		Зачет, вопросы 1-50
--	---	---	--	---------------------

## 2. Методика оценки этапов формирования компетенций в рамках дисциплины.

Промежуточная аттестация по **дисциплине** проводится в 1 семестре - в форме зачета, который направлен на оценку сформированности компетенций ОК.8, ОК.9, ОПК.2.

Зачет проводится в устной форме

Кроме того, сформированность компетенций проверяется при проведении мероприятий текущего контроля, указанных в таблице раздела 1.

В 1 семестре обязательным этапом текущей аттестации является расчетно-графическое задание (работа) (РГЗ). Требования к выполнению РГЗ, состав и правила оценки сформулированы в паспорте РГЗ.

Общие правила выставления оценки по дисциплине определяются балльно-рейтинговой системой, приведенной в рабочей программе учебной дисциплины.

На основании приведенных далее критериев можно сделать общий вывод о сформированности компетенций ОК.8, ОК.9, ОПК.2, за которые отвечает дисциплина, на разных уровнях.

### Общая характеристика уровней освоения компетенций.

**Ниже порогового.** Уровень выполнения работ не отвечает большинству основных требований, теоретическое содержание курса освоено частично, пробелы могут носить существенный характер, необходимые практические навыки работы с освоенным материалом сформированы не достаточно, большинство предусмотренных программой обучения учебных заданий не выполнены или выполнены с существенными ошибками.

**Пороговый.** Уровень выполнения работ отвечает большинству основных требований, теоретическое содержание курса освоено частично, но пробелы не носят существенного характера, необходимые практические навыки работы с освоенным материалом в основном сформированы, большинство предусмотренных программой обучения учебных заданий выполнено, некоторые виды заданий выполнены с ошибками.

**Базовый.** Уровень выполнения работ отвечает всем основным требованиям, теоретическое содержание курса освоено полностью, без пробелов, некоторые практические навыки работы с освоенным материалом сформированы недостаточно, все предусмотренные программой обучения учебные задания выполнены, качество выполнения ни одного из них не оценено минимальным числом баллов, некоторые из выполненных заданий, возможно, содержат ошибки.

**Продвинутый.** Уровень выполнения работ отвечает всем требованиям, теоретическое содержание курса освоено полностью, без пробелов, необходимые практические навыки работы с освоенным материалом сформированы, все предусмотренные программой обучения учебные задания выполнены, качество их выполнения оценено числом баллов, близким к максимальному.

Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Новосибирский государственный технический университет»  
Кафедра аэрогидродинамики  
Кафедра прочности летательных аппаратов

## Паспорт зачета

по дисциплине «Информатика», 1 семестр

### 1. Методика оценки

Зачет проводится в устной форме, по билетам. Билет формируется по следующему правилу: первый вопрос выбирается из диапазона вопросов 1-25, второй вопрос из диапазона вопросов 26-50 (список вопросов приведен ниже). В ходе зачета преподаватель вправе задавать студенту дополнительные вопросы из общего перечня (п. 4).

### Форма билета для зачета

НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ  
Факультет ФЛА

Билет № \_\_\_\_\_

к зачету по дисциплине «Информатика»

---

1. Кодирование (представление) данных в ЭВМ
2. Решение нелинейных уравнений.

Утверждаю: зав. кафедрой ПЛА \_\_\_\_\_ проф., Пустовой Н.В.  
(подпись) (дата)

### 2. Критерии оценки

- Ответ на билет для зачета считается **неудовлетворительным**, если студент при ответе на вопросы не дает определений основных понятий, не способен показать причинно-следственные связи явлений, допускает принципиальные ошибки, оценка составляет *25 баллов*.
- Ответ на билет для зачета засчитывается на **пороговом** уровне, если студент при ответе на вопросы дает определение основных понятий, может показать причинно-следственные связи явлений, допускает непринципиальные ошибки, например, вычислительные, оценка составляет *50 баллов*.
- Ответ на билет для зачета билет засчитывается на **базовом** уровне, если студент при ответе на вопросы формулирует основные понятия, законы, дает характеристику процессов, явлений, проводит анализ причин, условий, может представить

качественные характеристики процессов, не допускает ошибок, оценка составляет 75 баллов.

- Ответ на билет для зачета билет засчитывается на **продвинутом** уровне, если студент при ответе на вопросы проводит сравнительный анализ подходов, проводит комплексный анализ, выявляет проблемы, предлагает механизмы решения, способен представить количественные характеристики определенных процессов, приводит конкретные примеры из практики, не допускает ошибок и способен обосновать выбор метода решения задачи, оценка составляет 100 баллов.

### **3. Шкала оценки**

Зачет считается сданным, если сумма баллов по всем заданиям билета оставляет не менее 50 баллов (из 100 возможных).

В общей оценке по дисциплине баллы за зачет учитываются в соответствии с правилами балльно-рейтинговой системы, приведенными в рабочей программе дисциплины.

### **4. Вопросы к зачету по дисциплине «Информатика»**

1. Основные понятия информатики
2. Меры и единицы измерения информации
3. Системы счисления
4. Кодирование (представление) данных в ЭВМ
5. Основные понятия алгебры логики
6. Логические основы ЭВМ
7. Основные этапы развития вычислительной техники. Архитектура ЭВМ. Принципы работы вычислительной системы
8. Состав и назначение основных элементов персонального компьютера. Центральный процессор. Системные шины и слоты расширения
9. Запоминающие устройства: классификация, принцип работы, основные характеристики
10. Устройства ввода/вывода данных, их разновидности и основные характеристики
11. Виды программного обеспечения и их характеристики
12. Понятие и назначение операционной системы. Классификация программного обеспечения. Виды программного обеспечения и их характеристики
13. Файловая структура операционных систем
14. Основы машинной графики
15. Программное обеспечение обработки текстовых данных
16. Электронные таблицы
17. Формулы, работа со списками, создание диаграмм в MS Excel
18. Электронные презентации
19. Модели решения функциональных и вычислительных задач
20. Классификация и формы представления моделей
21. Методы и технологии моделирования
22. Информационные модели
23. Алгоритм и его свойства. Способы записи алгоритма
24. Базовые алгоритмические структуры
25. Типовые алгоритмы
26. Эволюция и классификация языков программирования
27. Трансляция, компиляция и интерпретация
28. Этапы решения задач на компьютерах
29. Структурное программирование
30. Объектно-ориентированные методы анализа и проектирования ПО
31. Общие понятия баз данных
32. Модели данных в информационных системах

33. Основные операции с данными
34. Базовые понятия ИИ
35. Компоненты вычислительных сетей
36. Принципы построения сетей. Сетевое оборудование.
37. Сервисы Интернета
38. Средства использования сетевых ресурсов. Браузеры. Почтовые программы
39. Защита информации в локальных и глобальных компьютерных сетях
40. Алгоритмы линейной, разветвляющейся, циклической структур.
41. Циклы с заданным числом повторений и итерационные. Вложенные циклические структуры.
42. Численное интегрирование. Метод прямоугольников.
43. Численное интегрирование. Метод трапеций.
44. Численное интегрирование. Метод Симпсона.
45. Определение длины кривой
46. Решение нелинейных уравнений.
47. Особенности выполнения Fortran-программы. Основные символы. Правила записи программ. Типы величин. Константы, переменные, массивы.
48. Особенности выполнения Fortran-программы. Операторы описания. Арифметические выражения. Арифметический оператор присваивания. Логические выражения. Логический оператор присваивания. Операторы управления.
49. Операторы цикла. Операторы ввода-вывода. Оператор FORMAT.
50. Программные единицы. Оператор-функция. Подпрограмма-функция. Подпрограмма.



## Паспорт расчетно-графического задания (работы)

по дисциплине «Информатика», 1 семестр

### 1. Методика оценки

В рамках расчетно-графического задания (работы) по дисциплине студенты должны выполнить задания в соответствии с индивидуальным вариантом.

При выполнении расчетно-графического задания (работы) студенты должны составить требуемые программы, протестировать и оформить результаты расчетов в виде отчета. Правила оформления: работы выполняются с помощью текстового процессора (желательно MS Word), параметры страницы: поля – левое – 2.5см., правое – 1см., верхнее и нижнее – 1.5см. Текст – 12пт., Times New Roman, межстрочный интервал – одинарный, выравнивание абзацев – по ширине, рисунки не должны иметь чёрный/тёмный фон, заполненность страницы информацией не должна быть менее 70% (без необходимости). Обязательные структурные части РГЗ: задание, решение. Оцениваемые позиции: корректность выполнения задания, соблюдение требований к оформлению РГЗ.

### 2. Критерии оценки

- Работа считается **не выполненной**, если выполнены не все части РГЗ(Р), оценка составляет 25 баллов.
- Работа считается выполненной **на пороговом** уровне, если допущены ошибки в расчетах и/или оформлении, оценка составляет 50 баллов.
- Работа считается выполненной **на базовом** уровне, если допущена одна ошибка или незначительно нарушены правила оформления, оценка составляет 75 баллов.
- Работа считается выполненной **на продвинутом** уровне, если работа выполнена без ошибок и соблюдены все правила оформления, оценка составляет 100 баллов.

### 3. Шкала оценки

В общей оценке по дисциплине баллы за РГЗ(Р) учитываются в соответствии с правилами балльно-рейтинговой системы, приведенными в рабочей программе дисциплины.

### 4. Примерный перечень тем РГЗ(Р)

Задача 1 найти область определения нескольких функций, рассчитать их значения на выбранном интервале и построить графики используя Fortran, MathCAD, Excel

Задача 2 Для заданной области на плоскости составить и протестировать программу для определения попадания заданной пользователем точки в эту область используя Fortran, MathCAD

Задача 3 составить программу расчета суммы заданного сходящегося бесконечного ряда используя Fortran, MathCAD

Задача 4 составить и протестировать программу обработки массивов в соответствии индивидуальным заданием

Задача 5 составить и протестировать программу, реализующую методы решения нелинейных уравнений (метод половинного деления, метод Ньютона, метод простых итераций)

Задача 6 составить и протестировать программу вычисления определенных интегралов численными методами прямоугольников, трапеций, Симпсона (парабол) и Гаусса.

## Лабораторная работа № 1

### ТАБУЛИРОВАНИЕ ФУНКЦИЙ

Цель работы. При табулировании и построении графиков функций получить начальные навыки программирования и работы на компьютере.

Задание. Получить таблицы шести функций, заданных аналитическими выражениями. Используя полученные таблицы и табличный процессор **Microsoft Excel**, построить графики функций. Варианты заданий приведены в таблице 1.

Для выполнения задания предлагается использовать распространенный при математических вычислениях алгоритмический язык Фортран (FORTRAN). Выбор этого языка обусловлен тем, что Фортран предназначен для быстрого счета в различных научно-технических приложениях. Именно на решение научно-технических задач нацелена подготовка специалистов на факультете летательных аппаратов.

### Введение

Выражение — это формула для получения значения. Выражения образуются из операндов и знаков операций, объединяемых по правилам синтаксиса алгоритмического языка. В качестве операндов выражения используются буквальные и именованные константы, переменные, указатели функций. В зависимости от типа возвращаемого результата выражения подразделяются на арифметические, логические, символьные и производного типа. В первой лабораторной работе рассмотрим правила составления арифметических выражений.

При записи арифметических выражений в Фортране используются следующие арифметические операции и функции (в порядке возрастания приоритета):

– , + — вычитание и сложение;

\* , / — умножение и деление;

\*\* — возведение в степень;

COS(x) —  $\cos(x)$

SIN(x) —  $\sin(x)$

TAN(x) —  $\operatorname{tg}(x)$

ATAN(x) —  $\operatorname{arctg}(x)$

ALOG(x) —  $\ln(x)$

ALOG10(x) —  $\lg(x)$

EXP(x) —  $e^x$

ABS(x) —  $|x|$

SQRT(x) —  $\sqrt{x}$

INT(x) — целая часть  $x$ .

Порядок выполнения операций одного и того же приоритета – слева направо (кроме возведения в степень). Для изменения вышеупомянутого порядка вычислений используются круглые скобки.

Далее рассмотрим пример выполнения следующего варианта:

1.  $y = 0.5 \cdot \sin(3x + 2);$

5.  $x = 2 \cdot \cos(t) \cdot \sqrt{|\cos(2t)|};$

2.  $y = 1 + \sqrt{12 - x^2 - 4x};$

$y = 2 \cdot \sin(t) \cdot \sqrt{|\cos(2t)|};$

3.  $y = 1 - 3^x;$

4.  $y = \frac{x}{(1 - (x/2))};$

6.  $\rho = \frac{16}{5 - 3 \cdot \cos(\varphi)}$

Следует остановиться на подготовке исходных данных для программы. Вначале обязательно устанавливают область определения функции. Для 1-ой функции областью определения является вся числовая ось от  $-\infty$  до  $+\infty$ . Но вследствие периодичности функции синус достаточно построить график в интервале  $0 \leq 3x + 2 \leq 2\pi$  или  $-\frac{2}{3} \leq x \leq \frac{2 \cdot (\pi - 1)}{3}$ . Видно также, что областью изменения функции является интервал  $-0.5 \leq y \leq 0.5$ . Каждая пара значений  $x$  и  $y$  должна быть записана в таблицу результатов счета.

Запишем алгоритм решения первой задачи.

1. Зададим тип и значение неизменяемому параметру  $\pi$ .
2. Опишем тип используемых в программе переменных.
3. Откроем файл для записи значений  $x$  и  $y$ .
4. Определим начальное  $a$  и конечное  $b$  значения переменной  $x$ .

5. Введем количество строк  $n$  для табулирования функции.
6. Вычислим шаг  $h$  цикла для изменения переменной  $x$ .
7. Организуем цикл с управляющей переменной  $i$  целого типа.
8. В цикле вычислим значения переменных  $x$  и  $y$ .
9. Напечатаем на дисплей значения  $x$  и  $y$ .
10. Повторим печать значений  $x$  и  $y$  в файл.
11. Закончим цикл.

В соответствии с алгоритмом составим Фортран-программу:

```

PROGRAM LAB1
REAL, PARAMETER :: PI=3.141593
REAL A, B, H, X, Y
INTEGER I, N
OPEN(1, FILE='LAB1.TXT')
A=-2./3.; B=2.*(PI-1.)/3.
PRINT *, ' N=?'; READ *, N
H=(B-A)/N
  DO I=0,N
    X=A+H*I
    Y=0.5*SIN(3.*X+2.)
    PRINT 2, X,Y; WRITE(1, 2) X,Y
  END DO
2 FORMAT (' X=',F8.4,3X,'Y=',F8.4)
END PROGRAM LAB1

```

Прокомментируем работу программы:

- вначале после ключевого слова PROGRAM объявляется имя программы, которое обязательно должно начинаться с буквы латинского алфавита или символа \$ (в общем случае ключевое слово PROGRAM с именем программы может отсутствовать);
- первый оператор задает тип и значение именованной константе PI;

- затем идут два оператора, описывающие тип используемых в программе переменных;
- оператор OPEN назначает логический номер 1 файлу «LAB1.TXT». Указанный файл создается в папке проекта либо предварительно, либо в процессе работы программы в момент выполнения оператора OPEN;
- в следующей строке через точку с запятой записаны два оператора, которые вычисляют значения переменных A и B (обратите внимание на десятичные точки в неименованных константах вещественного типа);
- оператор PRINT выводит на дисплей в свободном формате символьную константу, которая заключена в апострофы. Эта последовательность символов является подсказкой для пользователя;
- оператор READ требует ввода с клавиатуры значения целой переменной N. После нажатия клавиши [Enter] переменной N будет присвоено набранное значение;
- следующий оператор вычисляет значение переменной H ;
- оператор цикла DO организует повторение вычислений в теле цикла необходимое число раз при изменении управляющей переменной I от 0 до N с шагом 1 (шаг равен 1 по умолчанию, т.к. его значение не указано);
- операторы в следующих двух строках обеспечивают вычисление арифметических выражений, записанных справа от знака равенства. Результаты присваиваются переменным X и Y;
- оператор PRINT работает под управлением оператора FORMAT с меткой 2. Он выводит на экран сначала символьную константу X=, которая в программе заключена в апострофы, затем числовое значение переменной X по дескриптору F8.4. Далее через три позиции выводится символьная константа Y=, и значение Y также по дескриптору F8.4 , где 8 – общее число позиций для выводимого числа, а 4 – количество позиций для цифр после десятичной точки. В общее число позиций включается знак отрицательного числа «-» и десятичная точка (ноль перед десятичной точкой, как правило, не печатается);

- оператор WRITE повторяет печать значений X и Y в файл «LAB1.TXT» с логическим номером 1 под управлением оператора FORMAT с меткой 2;
- предпоследний оператор END DO ограничивает тело цикла;
- последний оператор END PROGRAM LAB1 ограничивает программный модуль, заканчивая его (LAB1 или PROGRAM LAB1 могут быть опущены). Если имя программы не объявлено, то оно не может присутствовать и в операторе END.

### Краткая инструкция по работе в среде программирования

Для запуска составленной программы на счет необходимо прежде всего создать проект. Для этого предлагаем использовать среду **Microsoft Developer Studio (MDS)**, в которой разрабатываются приложения на языке **Microsoft Fortran PowerStation (FPS)** версии 4.0, либо более позднюю «мастерскую разработчика» **Developer Studio (DS)**, в которой программирование осуществляется на языке **Compaq Visual Fortran (CVF)** версии 6.1. Для входа в среду **MDS** или **DS** необходимо после нажатия кнопки «Пуск» выбрать соответствующий установленной версии пункт меню “Программы”, либо другим способом — дважды щелкнуть по соответствующей иконке (**MDS** или **DS**) на рабочем столе.

Далее (при работе в **MDS**) выполним следующую цепочку действий: **File – New – Project Workspace – OK – QuickWin Application – ввести имя проекта – задать расположение проекта на диске – Create**. В качестве места расположения проекта студентам рекомендуется использовать папку с именем группы специально созданную в папке FLA на жестком диске C: компьютера. После нажатия **Create** будет создана папка с именем проекта. Для сохранения конфигурации проекта можно использовать меню **File – Save All**. В папке с именем проекта будут размещены файлы с расширениями **.mak** и **.mdp**, фиксирующие структуру проекта. Желательно запомнить имя и расположение проекта для того, чтобы при необходимости можно было найти и открыть созданный проект.

При работе с более поздней версией **DS** для создания проекта необходимо выполнить другую цепочку действий: **File – New – выбрать закладку *Projects – Fortran Standard Graphics or QuickWin Application* – ввести имя проекта – задать расположение проекта на диске – OK**. В появившемся окне нужно выбрать кнопку **QuickWin (multiple windows)** и нажать **Finish**. В результате этих действий создается папка с именем проекта. В этой папке будут размещены файлы с расширениями **.dsp**, **.dsw** и **.opt**, фиксирующие структуру проекта. Сам же проект отобразится в окне с закладкой **File View**.

Заметим, что уже существующий проект открывается по цепочке **File – Open Workspace – Выбор файла с расширением .mdp или .dsw (в зависимости от версии) – Open**, а закрывается командами **File – Close Workspace**.

Создадим теперь файл, в который будет занесен текст программы, выполнив действия: **File – New – Text File – OK**. Желательно еще до набора текста программы присвоить этому файлу имя в режиме **File – Save** или **File – Save As**. Сохраненный файл, который должен иметь расширение **.f90**, нужно включить в проект. Для этого при нахождении курсора мышки в правом окне нужно нажать правую клавишу мышки, в появившемся контекстном меню выбрать пункт **Insert File into Project** и щелкнуть левой клавишей мышки на имени проекта. Далее наберем в правом окне текст программы. В процессе набора ключевые слова Фортрана и имена встроенных функций отображаются синим цветом, а комментарии — зеленым. Начальный отступ предыдущей строки наследуется, что облегчает набор структурированных программ в свободном формате. При наборе больших программ желательно периодически сохранять файл, записывая его на диск.

Если файл с программой уже имеется, то добавить его в проект можно выполнив действия: **Insert – Files into Project – Выбор файла – Add**. Чтобы отобразить в правом окне содержимое файла, достаточно с помощью левой клавиши мышки дважды щелкнуть по нему в окне **FileView** (в папке проекта). Для удаления файла из открытого проекта достаточно выбрать этот файл в окне **FileView** и нажать клавишу [Del].

При работе с более поздней версией **DS** для создания файла, в который будет занесен текст программы, необходимо выполнить: **File – New – Выбрать закладку Files – выбрать Fortran Free Format Source File – Активизировать опцию (поставить «галочку») Add to project – Задать имя файла – ОК**. Созданный файл должен иметь расширение **.f90**.

Если же файл с программой уже существует, то для добавления его в проект нужно: в окне **FileView – выбрать папку Source Files – нажать на правую кнопку мышки – Add Files to Folder – выбрать тип файлов – выбрать необходимые файлы – ОК**. Чтобы отобразить на экране содержимое файла, достаточно с помощью левой клавиши мышки дважды щелкнуть по нему в окне **FileView** (в папке **Source Files**).

Для перевода программы на язык машинного уровня выполним компиляцию проекта: меню **Build – Compile** и исправим обнаруженные ошибки, сообщения о которых появятся в нижнем окне экрана. При отсутствии ошибок программа переводится с языка Фортран на язык машинного (объектного) уровня и создается объектный модуль с расширением **.obj**.

Далее с помощью меню **Build – Build** построим исполняемый файл с расширением **.exe**, в котором объединяются объектные коды главной программы и вызываемых ею вспомогательных процедур, в т.ч. встроенных.

Исполняемый файл запустим на счет: **Build – Execute**. Результат выводится в специальное рабочее окно. Отметим, что кириллические тексты на экране воспроизводятся в искаженном виде. Фрагменты содержимого окна результатов можно переносить в другие файлы через системный буфер. Прекратить счет можно нажатием комбинации клавиш [Ctrl + C]. Если программа закончила работу, то можно выйти из окна сразу и вернуться к программе или поработать с результатами средствами операционной системы **Windows** и затем закрыть окно.

Компиляцию, построение исполняемого модуля и запуск программы можно также выполнить с помощью кнопок панели инструментов или клавишных комбинаций, имеющих в меню **Build**.



### Получение таблиц остальных заданных функций

Во 2-ой задаче область определения функции устанавливается исходя из того, что подкоренное выражение должно быть неотрицательно. Следовательно,  $12 - x^2 - 4x \geq 0$ . Решая квадратное уравнение, получим область определения:  $-6 \leq x \leq 2$ . При исполнении программы все значения аргумента следует брать только из этой области, так как вне ее функция не определена. Следовательно, в программе для 2-ой задачи требуется изменить значения переменных *A* и *B* в третьей строке, набрав вместо старых операторов новые

*A*=-6.; *B*=2.

И в соответствии с заданием изменяется оператор вычисления переменной *y*

*Y*=1.+SQRT(12.-*X*\*\*2-4.\**X*)

Указанные изменения в программе лучше оформить в отдельном файле, который можно получить из прежнего, выполнив ***File – Save As*** и набрав имя нового файла. Затем следует выполнить удаление из проекта старого файла, включение в проект нового файла, компиляцию, построение загрузочного модуля и запуск программы.

Аналогично проводятся вычисления для 3-ей и 4-ой функций, для которых также лучше оформить отдельные файлы.

В 5-ой задаче функция задается параметрически, поэтому здесь сначала в цикле вычисляется переменная *T*, соответствующая параметру функции, а затем переменные *X* и *Y*. В результате операторы присваивания принимают вид:

*T*=*A*+*H*\**I*

*X*=2.\*COS(*T*)\*SQRT(ABS(COS(2.\**T*)))

*Y*=2.\*SIN(*T*)\*SQRT(ABS(COS(2.\**T*)))

В 6-ой задаче функция описывается полярными координатами с радиус-вектором  $\rho$  из начала декартовых координат и углом  $\varphi$ , который отсчитывается от оси абсцисс против часовой стрелки. Целесообразно перейти от полярных координат к декартовым координатам по формулам:

$$x = \rho \cdot \cos(\varphi); \quad y = \rho \cdot \sin(\varphi).$$

С использованием переменных  $r = \rho$ ,  $f = \varphi$  для вычисления значений  $x$  и  $y$  новую программу лучше сделать из предыдущей, в которой следует заменить переменную  $T$  на переменную  $F$ , а также изменить два оператора присваивания для переменных  $X$ ,  $Y$  и перед ними добавить еще один, соответствующий заданной функции:

$$R=16./(5.-3.*\cos(F))$$

$$X=R*\cos(F); Y=R*\sin(F)$$

В рассмотренном примере угол  $\varphi$  необходимо задавать в интервале от 0 до  $2\pi$ .

В общем случае область определения функции находится из условия  $\rho \geq 0$ .

### Построение графиков функций в табличном процессоре Microsoft Excel

Перед началом работы необходимо осуществить запуск **Excel**, щелкнув мышью на кнопке <Пуск> и выполнив команду **Программы / Microsoft Excel**. Пустая электронная таблица, открываемая **Excel** сразу после его запуска, по умолчанию имеет имя **Book1 (Книга1)**.

Ввод данных. Ввод данных осуществляется в активную ячейку. Активной называется ячейка, в которой в данный момент времени расположен табличный (или окошечный) курсор. Перемещение курсора осуществляется с помощью клавиатуры или с помощью мыши. Введите данные в пустую электронную таблицу в следующем порядке:

- сделайте активной ячейку A1 и введите текст «Функция 1». Обратите внимание, что вводимая информация дублируется в строке формул. Для окончания ввода информации в ячейку необходимо или нажать клавишу **Enter**, или перевести курсор в другую ячейку.
- Сделайте активной ячейку A2 и введите первое значение  $x$ , затем в активную ячейку A3 — второе значение  $x$  и так далее весь диапазон изменения  $x$ . Например, если количество значений  $x$  равно 11, то последнее значение  $x$  будет записано в ячейку A12. Ввод значений  $x$  можно существенно упростить, если переменная  $x$  изменяется с постоянным шагом. Для этого нужно выделить (маркировать) хотя бы две ячейки, содержащие значения  $x$ ,

отличающиеся на заданный шаг. Выделение осуществляется буксировкой мышки (при нажатой левой клавиши) по выбранным ячейкам. Затем нужно отпустить левую клавишу и поместить курсор мышки в правый нижний угол выделенной области так, чтобы появился крестик черного цвета. Далее следует снова нажать левую клавишу мышки и продолжить буксировку вниз до появления крайнего значения  $x$  из заданного диапазона, после чего можно отпустить левую клавишу мышки.

- Переведите курсор мыши в ячейку B2 и введите в нее первое значение  $y$  рассматриваемой функции, затем в ячейку B3 введите второе значение  $y$  и так далее пока не будет введено последнее значение функции в ячейку B12.

Построение графиков. Во-первых, необходимо выделить область таблицы, содержащую данные, по которым нам необходимо построить график. В нашем случае это ячейки, начиная с A2 до B12. Выделение группы ячеек осуществляется буксировкой мыши. Затем нужно использовать команду построения диаграммы «Вставка / Диаграмма» или соответствующую пиктограмму. Далее:

- выбираем тип диаграммы «Точечная» и её вид;
- проверяем диапазон адресов маркированных областей данных;
- вводим название диаграммы и осей;
- выбираем режим «построение диаграммы на имеющемся листе».

При необходимости осуществляется корректировка диаграммы. Для этого нужно активизировать диаграмму щелчком левой кнопки мыши, щелкнуть правой кнопкой мыши в области диаграммы на свободном от её элементов месте и в открывшемся контекстном меню выбрать нужную команду или элемент.

Сохранение документа на диске. Выполните команду «Файл / Сохранить как». В открывшемся окне в поле ввода «Имя файла» введите название файла «Lab1.xls». Обратите внимание, в какой папке будет сохранен документ (при необходимости откройте другую папку). Наконец, нажмите командную кнопку «Сохранить». Аналогично внесите данные и постройте графики остальных пяти функций, используя свободные ячейки по своему усмотрению.

**Таблица 1. Варианты заданий к лабораторной работе № 1**

№ п/п	1	2	3	4	5	6
1	$y = 2 \sin(x/3)$	$y =  x+1  -  x-1 $	$y = 2 + \sqrt{4 - x^2 - 6x}$	$y = -2 + \frac{2}{3}\sqrt{x^2 - 9}$	$x = \arcsin(t)$ $y = \operatorname{arctg}(t)$	$\rho = \varphi$
2	$y = 2 \sin(3x + \pi)$	$y = 2 x  - x + 1$	$y = -1 + \sqrt{2 - x^2 - 5x}$	$y = \frac{x+1}{x+2}$	$x = \cos(t)$ $y = \sin^2(t)$	$\rho = \frac{\pi}{\varphi}$
3	$y = \operatorname{tg}(2x+1)$	$y = x -  x $	$y = 2 - \sqrt{4 - x^2 - 6x}$	$y = -3 \cdot \sqrt{(x+1)^2 + 1}$	$x = \cos(2t)$ $y = \cos^2(t)$	$\rho = e^{\pi/\varphi}$
4	$y = 2 \cos(3x+1)$	$y = \ln(\sqrt{x-4})$	$y = \frac{1}{2} - \sqrt{x^2 - 2y - \frac{1}{4}}$	$y = 1 + \frac{7}{3}\sqrt{3 + x^2 + 4x}$	$x = 1$ $y = \sin(t)$	$\rho = 2\varphi$
5	$y = -\sin(\frac{x}{2} - 1)$	$y = \sqrt{x^3}$	$y = -1 - \sqrt{2 - x^2 - 5x}$	$y = 3 - \frac{4}{3}\sqrt{x^2 + 9}$	$x = \sqrt{\cos(t)}$ $y = \sqrt{\sin(t)}$	$\rho = \sec^2\left(\frac{\varphi}{2}\right)$
6	$y = 2 \operatorname{tg}(2x) $	$y = \ln(x^2)$	$y = 1 + \frac{3}{4}\sqrt{5 - x^2 - 2x}$	$y = 1 - \frac{4}{3}\sqrt{1 + x^2 + 4x}$	$x = \sin(t)$ $y = \ln(\sin(t))$	$\rho^2 \cos(2\varphi) = 1$
7	$y = \frac{1}{2} \sin 2x $	$y = 2 \cdot \ln(x)$	$y = 2 - \frac{5}{2}\sqrt{8 - x^2} + x$	$y = 1 + \frac{2}{3}\sqrt{x^2 - x + 5}$	$x = \arcsin(t)$ $y = t^2$	$\rho = \frac{2}{1 - \cos(\varphi)}$
8	$y = \operatorname{tg}\left \frac{x}{2}\right $	$y =  x  - 1$	$y = 1 - \frac{2}{3}\sqrt{5 - x^2 - x}$	$y = 2 + \frac{3}{4}\sqrt{x^2 - x - 6}$	$x = 2^t$ $y = 2^{2t}$	$\rho = \operatorname{tg}\left(\frac{\varphi}{2}\right)$

Продолжение табл. 1

№ п/п	1	2	3	4	5	6
9	$y = 2 \sin(\frac{\pi}{3} - x)$	$y = x +  x $	$y = 1 + \frac{1}{7} \sqrt{8 - x^2} - x$	$y = 2 + \frac{2}{5} \sqrt{x^2 + 25}$	$x = 3^t$ $y = 9^t - 3^t$	$\rho = \operatorname{cosec}^2(\frac{\varphi}{2})$
10	$y =  \cos(3x) $	$y = 2 x - 3  + 5x$	$y = 3 + \frac{3}{5} \sqrt{9 - x^2} + x$	$y = 2 - \frac{3}{4} \sqrt{x^2 - x - 3}$	$x = (t + 1/t)/2$ $y = (t - 1/t)/2$	$\rho = \varphi^2$
11	$y = \operatorname{tg}(\frac{x}{2} - 1)$	$y = -\frac{3}{2}x + 1$	$y = 1 + \frac{1}{2} \sqrt{5 - 3x^2} - x$	$y = 7 - \frac{3}{2} \sqrt{x^2 - 6x + 8}$	$x = 5^{t+1}$ $y = 5^{2t+3}$	$\rho = \sin(\frac{\varphi}{2})$
12	$y = \frac{3}{2} \cos(\frac{x}{2} + 1)$	$y = 2 x  + 3$	$y = -1 + \frac{1}{3} \sqrt{x^2 + 3x}$	$y = 1 + \frac{3}{4} \sqrt{x^2 + 4x + 9}$	$x = 2^{\sin(t)}$ $y = \sin(t)$	$\rho^2 \cos(2\varphi) = 9$
13	$y = \frac{2}{3} \sin(3x + 1)$	$y = \frac{1}{7}(x + 1) - 2$	$y = 2 + \frac{2}{3} \sqrt{4 - x^2} + x$	$y = 9 - 2\sqrt{x^2 + 4x + 8}$	$x = \sqrt{t}$ $y = 1/t$	$\rho = \frac{1}{2 + \cos(\varphi)}$
14	$y =  \operatorname{tg} 2x  $	$y = \sqrt{x^2 - 4x + 4}$	$y = 3 - \frac{3}{5} \sqrt{9 + 5x^2} + x$	$y = 1 - \frac{3}{4} \sqrt{x^2 + 4x + 1}$	$x = \cos(t)$ $y = 1/\sin(t)$	$\rho = \frac{2}{\cos(\varphi)}$
15	$y = \operatorname{tg}(1 - x)$	$y = \sqrt{x^2 - 6x + 9}$	$y = 1 - \frac{1}{2} \sqrt{6 - x^2} - 2x$	$y = 5 - \frac{4}{3} \sqrt{x^2 + 4x - 2}$	$x = \sec(t)$ $y = \cos(t)$	$\rho = \frac{2}{1 + \cos(\varphi)}$
16	$y = -\frac{2}{7} \sin(x + 1)$	$y = \frac{1}{5}(x - 1) + 1$	$y = 1 - \frac{1}{3} \sqrt{-5x^2 + 3x}$	$y = 2 + \frac{4}{3} \sqrt{x^2 + 2x - 1}$	$x = \cos(t)$ $y = \cos(2t)$	$\rho = \frac{8}{3 - \cos(\varphi)}$

Продолжение табл. 1

№ п/п	1	2	3	4	5	6
17	$y = -3\sin(5x)$	$y = \frac{1}{5}x - 1$	$y = 2 - \frac{2}{3}\sqrt{9 - x^2} + x$	$y = \frac{1}{x} - \frac{1}{2}$	$x = \operatorname{tg}(t)$ $y = \frac{2}{1 + \cos(t)}$	$\rho = 1 + \cos(\varphi)$
18	$y =  \operatorname{ctg}(3x) $	$y = \sqrt{(x-1)^2 + 3}$	$y = 1 + \frac{7}{3}\sqrt{x^2 - 4x + 5}$	$y = 2 - \frac{4}{3}\sqrt{x^2 + 2x - 8}$	$x = \sin(\frac{\pi}{2} + t)$ $y = \sin(\pi + t)$	$\rho^2 = 36\cos(\varphi)$
19	$y = 1 - \cos(3x)$	$y = 1 + \sqrt{x^2 - x + 1}$	$y = 2 - \frac{3}{7}\sqrt{x^2 + 2x + 4}$	$y = \frac{2x}{3 - x}$	$x = \cos(t)$ $y = \pi \cdot \sin(t)$	$\rho = 1 - \cos(\varphi)$
20	$y = \frac{1}{3} \cos(2x) $	$y = 1 + \sqrt{x^2 - x + 4}$	$y = 1 + \frac{4}{3}\sqrt{-6x - x^2}$	$x = 1 + \frac{3}{2}\sqrt{y^2 + 4y + 8}$	$x = 1 + \operatorname{tg}(t)$ $y = \operatorname{ctg}^2(t)$	$\rho = \cos(2\varphi)$
21	$y = \operatorname{ctg} 3x - 3 $	$y =  x  -  x + 2 $	$y = 1 - \frac{7}{3}\sqrt{x^2 - 4x + 5}$	$y = \frac{-x + 4}{x - 1}$	$x = \operatorname{cosec}(t)$ $y = \operatorname{tg}^2(t)$	$\rho = \sin^3 \varphi$
22	$y =  \operatorname{tg}(x)  + 2$	$y = 3 - \sqrt{x^2 - x + 1}$	$y = 7 + \frac{2}{5}\sqrt{6 - 6x - x^2}$	$y = 1 - \frac{3}{2}\sqrt{x^2 + 4x + 8}$	$x = \operatorname{cosec}(t)$ $y = \sin(t)$	$\rho = \sin^2(\varphi)$
23	$y =  \sin(x) - 1 $	$y = \frac{3}{2}x + \frac{1}{2}$	$y = 2 + \frac{3}{7}\sqrt{x^2 + 2x + 8}$	$y = \frac{x - 1}{2x - 5}$	$x = \sec(t)$ $y = \ln(\cos(t))$	$\rho = 4\sin(\varphi)$
24	$y = \frac{1}{3}\sin\left \frac{x}{2}\right $	$y = \ln x $	$y = 2 - \sqrt{-5 - 6x - x^2}$	$y = 2 + \frac{2}{3}\sqrt{x^2 - x + 9}$	$x = \cos(t)$ $y = \lg(\sec(t))$	$\rho = \cos(\varphi) + \varphi$

Окончание табл. 1

№ п/п	1	2	3	4	5	6
25	$y =  \operatorname{tg}(3x + 2) $	$y =  \ln(x - 1) $	$y = 5 + \frac{1}{4}\sqrt{8 + 2x - x^2}$	$y = \frac{1 - x}{1 + x}$	$x = t - \sin(t)$ $y = 1 - \cos(t)$	$\rho = \frac{1}{2} + \sin(\varphi)$
26	$y = \operatorname{ctg}(x/2) - 1$	$y = e^{ x  - 1}$	$y = 5 + \sqrt{40 - 6x - x^2}$	$y = 2 - \frac{2}{3}\sqrt{x^2 - x + 8}$	$x = \cos^3(t)$ $y = \sin^3(t)$	$\rho = \sin(\varphi - \frac{\pi}{4})$
27	$y = \operatorname{ctg}(2x - 1)$	$y = e^{-x^2}$	$y = 3 + \sqrt{5 - x^2 - 4x}$	$y = \frac{3x - 5}{x - 1}$	$x = 5\cos^2(t)$ $y = 3\sin^2(t)$	$\rho = 1 + 2\cos(\varphi)$
28	$y = \left  \sin\left(\frac{x}{2} - 1\right) \right $	$y = \ln x + 1 $	$y = -5 - \sqrt{7 - x^2 - 4x}$	$y = 1 + \frac{4}{3}\sqrt{4 + x^2 + 4x}$	$x = 2\sec^2(t)$ $y = \sin(2t)$	$\rho = 1 - \sin(\varphi)$
29	$y = \frac{1}{2}\sin 3x + 2 $	$y = 1 - 3^x$	$y = 1 + \sqrt{12 - x^2 - 4x}$	$y = \frac{x}{1 - x/2}$	$x = \cos(t)$ $y = \sin(t)$	$\rho = \frac{8}{3 - \cos(\varphi)}$
30	$y = 2 + \operatorname{tg}\left(\frac{x}{3} + 1\right)$	$y = \left(\frac{1}{3}\right)^{ x - 1 }$	$y = -2 + \sqrt{-x^2 - 10x}$	$x = 1 - \frac{4}{3}\sqrt{4 + 4y + y^2}$	$x = t^2 / (1 + t^2)$ $y = t / (1 + t^2)$	$\rho = \cos^2\left(\frac{\varphi}{2}\right)$

## Лабораторная работа № 2

### ПРОГРАММИРОВАНИЕ ВЕТВЯЩИХСЯ ВЫЧИСЛИТЕЛЬНЫХ ПРОЦЕССОВ

Цель работы. Получить навыки программирования ветвящихся вычислительных процессов. Изучить условные операторы, переменные логического типа, логические отношения и операции.

Постановка задачи. Пусть на плоскости  $x, y$  задана область  $D$ . Требуется определить «попадает» ли точка с координатами  $(x, y)$  в область  $D$  или нет. Если попадает, то необходимо вычислить функцию  $z = f_1(x, y)$  и переменной  $M$ , значение которой является признаком «попадания» или «непопадания», присвоить значение  $M=1$ . В противном случае вычислить  $z = f_2(x, y)$  и положить  $M=2$ .

В каждом варианте задания область  $D$  можно разбить на две подобласти:  $D_1, D_2$ . Подобласть  $D_1$  ограничена прямыми линиями, уравнения которых нужно составить. При этом удобно использовать уравнение прямой в отрезках на осях:  $\frac{x}{a} + \frac{y}{b} = 1$ . Подобласть  $D_2$  ограничена кривой второго порядка, уравнение которой указано в задании. Варианты заданий приведены в таблице 2.

#### Пояснение задачи

1. Для заданного варианта из таблицы нарисовать область  $D$  и выписать функции  $f_1(x, y)$  и  $f_2(x, y)$ . Записать уравнения, определяющие границы области, и подготовить исходные данные (координаты точек) так, чтобы часть точек находилась внутри области  $D$ , а часть — вне ее (см. рис. 1). При этом для каждой четверти координатной плоскости рекомендуется выбрать по две точки.



2. Создать на компьютере проект для решения поставленной задачи, составить и отладить программу.
3. Для выбранных точек получить и записать результаты счета.

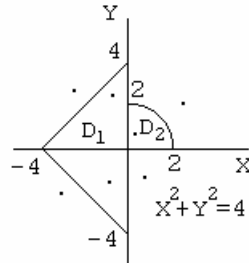


Рис. 1. Область  
на плоскости  $x, y$

Рассмотрим один из вариантов решения задачи. Пусть область  $D$  имеет вид, показанный на рисунке 1, а функции  $f_1$  и  $f_2$  заданы как

$$f_1(x, y) = \ln(|x + y| + 1);$$

$$f_2(x, y) = e^{\frac{1}{|x| + |y|}}.$$

Прямой  $x=0$  разобьем область  $D$  на две подобласти:  $D_1$  и  $D_2$ . Уравнения прямых, ограничивающих подобласть  $D_1$ , будут:

во второй четверти  $y = x + 4$ ; в третьей  $y = -x - 4$ .

Уравнение окружности с центром в начале координат, ограничивающей подобласть  $D_2$  при  $x > 0$  и  $y > 0$ , имеет вид  $x^2 + y^2 = 4$ .

Теперь можно записать, что некоторая точка  $(x, y)$  «попадает» в область  $D$ , если она «попадает» в область  $D_1$  или  $D_2$ , то есть

если  $x \leq 0$  и  $y \leq x + 4$  и  $y \geq -x - 4$  или  $x > 0$  и  $y \geq 0$  и  $x^2 + y^2 \leq 4$ .

Точка  $(x, y)$  «не попадает» в область  $D$ , если  $x \leq 0$  и  $y > x + 4$  или  $x \leq 0$  и  $y < -x - 4$  или

$x > 0$  и  $y < 0$  или  $x > 0$  и  $x^2 + y^2 > 4$ .

При выполнении задания можно проводить анализ как «попадания», так и «непопадания» точки в область  $D$ .

При составлении программы на Фортране для анализа «попадания» или «непопадания» точки в область  $D$  можно составить только одно логическое выражение с использованием следующих логических отношений и операций:

отношения:

(в свободном и фиксированном форматах)

$<$  — .LT. — меньше,

$\leq$  — .LE. — меньше или равно,

$>$  — .GT. — больше,

$\geq$  — .GE. — больше или равно,

$=$  — .EQ. — равно,

$\neq$  — .NE. — не равно.

операции:

(в порядке убывания приоритета)

.NOT. — отрицание,

.AND. — конъюнкция («и»),

.OR. — дизъюнкция («или»),

.EQV. — эквивалентность,

.NEQV. — неэквивалентность.

Результатом вычисления отношений является одно из двух логических значений .TRUE. — истина или .FALSE. — ложь. С помощью логических операций можно создавать сложные логические выражения, которые также могут принимать только два значения .TRUE. или .FALSE.

Например, точка  $(x, y)$  «попадает» в область  $D$  (см. рис. 1), если логическое выражение

$X \leq 0 \text{..AND.} Y \leq X + 4 \text{..AND.} Y \geq -X - 4 \text{..OR.} X > 0 \text{..AND.} Y \geq 0 \text{..AND.} X^2 + Y^2 \leq 4$ .

принимает значение .TRUE. — истина.

Ветвящиеся вычислительные процессы в Фортране можно запрограммировать с помощью следующих условных операторов IF.

**Логический оператор IF** имеет вид

IF(L) S

Здесь: L – логическое выражение, S – исполняемый оператор.

Если логическое выражение L истинно, то выполняется оператор S, если оно ложно, то выполнится следующий за IF оператор, а S не выполняется.

**Блочный оператор IF** имеет вид

```

IF(L) THEN
    < блок операторов 1 >
ELSE
    < блок операторов 2 >
END IF

```

Правило выполнения: если логическое выражение L истинно, то выполняется < блок операторов 1 >, а если оно ложно, то выполняется < блок операторов 2 >.

При отсутствии альтернативы < блок операторов 2 > вместе с ELSE опускается.

При необходимости дополнительных условий используется общий вид блочного оператора IF, который имеет вид

```

IF(L) THEN
    < блок операторов 1 >
ELSE IF(L+) THEN
    < дополнительный блок операторов >
ELSE
    < блок операторов 2 >
END IF

```

Здесь: L и L+ – основное и дополнительное логические выражения.

Правило выполнения: если логическое выражение L истинно, то выполняются < блок операторов 1 >; если выражение L ложно, а L+ истинно, то выполняются < дополнительный блок операторов >; если выражение L ложно и L+ ложно, то выполняются < блок операторов 2 >. Дополнительных условий, соответствующих им логических выражений и блоков операторов может быть несколько.

Приведем пример Фортран-программы, в которой для анализа «попадания» или «непопадания» точки используется логическая переменная C. Этой переменной присваивается значение логического выражения. Если C=.TRUE., то в блочном операторе IF (C) THEN выполняется первый блок операторов, а если C=.FALSE., то второй блок (стоящий после ELSE).

```

PROGRAM LAB2
LOGICAL C

```

```

REAL X, Y, Z
INTEGER I, M
DO I=1,8
    PRINT *, ' X=?, Y=? '; READ '(2F7.3)', X,Y
    C=X<=0..AND.Y<=X+4..AND.Y>=-X-4..OR.&
    &X>0..AND.Y>=0..AND.X**2+Y**2<=4.
    IF(C) THEN
        PRINT *, ' YES '
        Z=F1(X,Y); M=1
    ELSE
        PRINT *, ' NO '
        Z=F2(X,Y); M=2
    END IF
    PRINT 2, X,Y,Z,C,M
END DO
2 FORMAT (' X=',F7.3,3X,'Y=',F7.3,3X,'Z=',E11.4,3X,'C=', L2,3X,'M=',I2/)
END PROGRAM LAB2

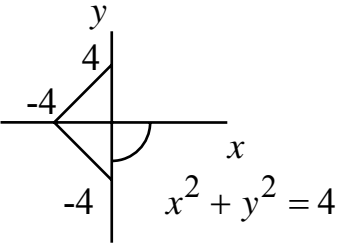
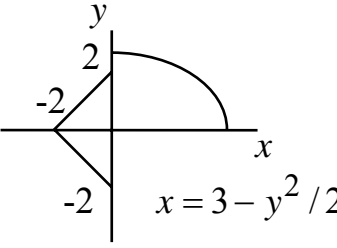
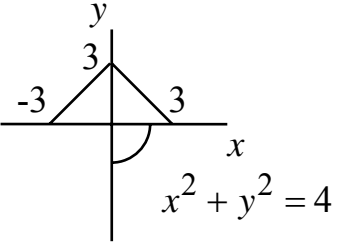
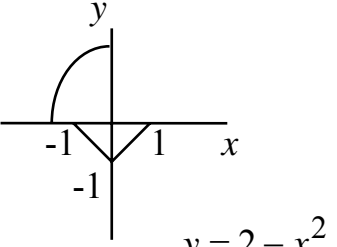
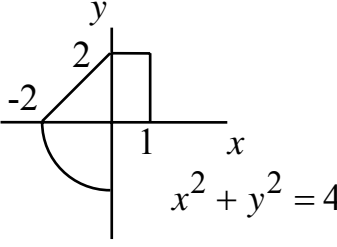
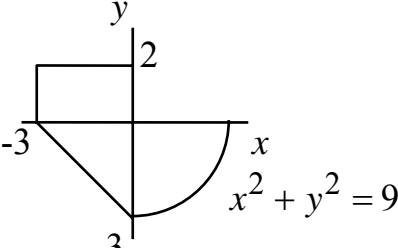
FUNCTION F1(X,Y)
F1=ALOG(ABS(X+Y)+1.)
END FUNCTION F1

FUNCTION F2(X,Y)
F2=EXP(1./(ABS(X)+ABS(Y)))
END FUNCTION F2

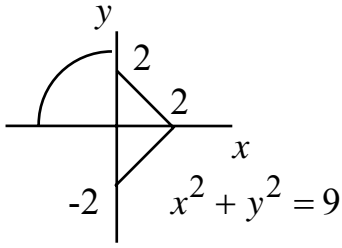
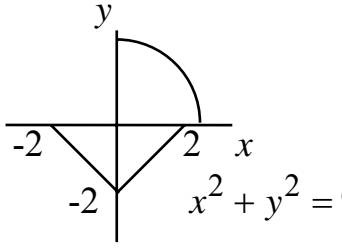
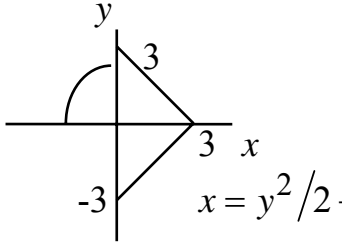
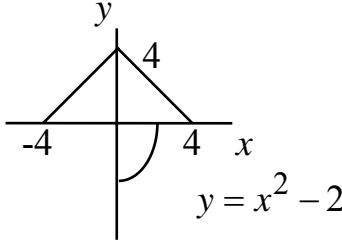
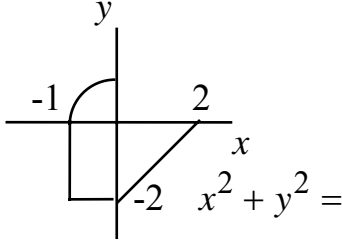
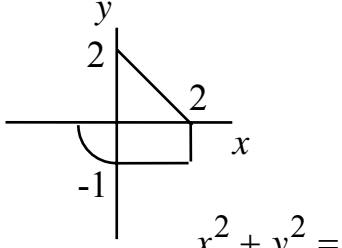
```

После окончания программы описаны внутренние процедуры-функции F1 и F2 .  
Эта программа для своего завершения требует ввести координаты 8 точек.

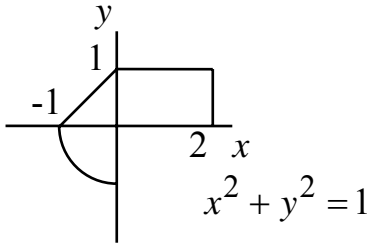
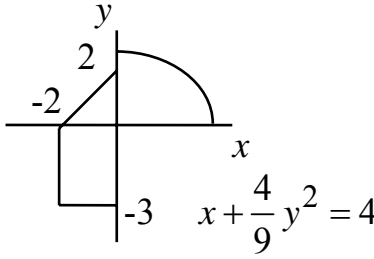
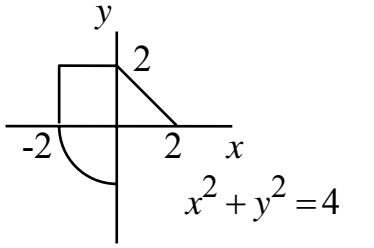
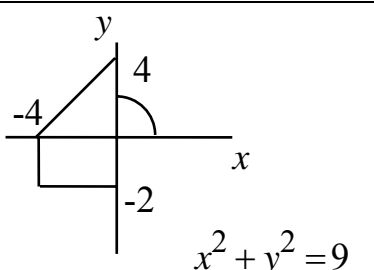
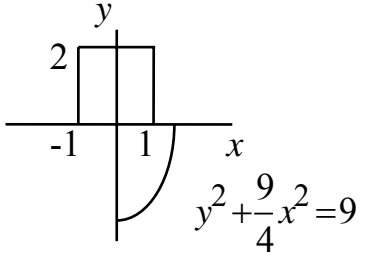
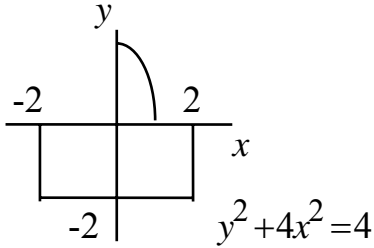
**Таблица 2. Варианты заданий к лабораторной работе № 2**

№ п/п	$f_1(x, y)$	$f_2(x, y)$	Область
1	$\operatorname{tg}(x) + \ln(y^2)$	$e^{\sin^2(x) + \cos^2(y)}$	
2	$\frac{\sin(x)}{1 - \cos(y)}$	$e^{\operatorname{arctg}(x^2 + y^2)}$	
3	$x^2 + \operatorname{tg}^2(y)$	$\ln(\sqrt{x^4 + 2y^2})$	
4	$\sqrt{ x \sin(y) }$	$\ln\left(\frac{1}{x^2} + \frac{1}{y^2}\right)$	
5	$e^{2\sin(x) + \cos(y)}$	$e^{\left(\frac{1}{x} + \frac{1}{y}\right)}$	
6	$x + \sin(y)$	$\operatorname{tg}(x^2 + y^2)$	

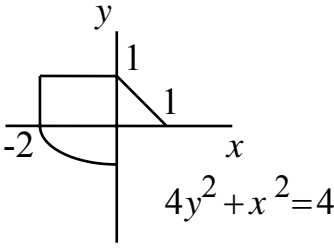
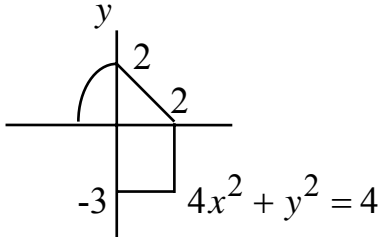
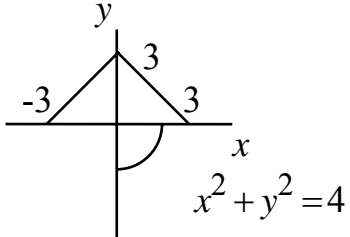
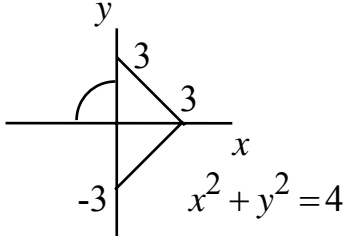
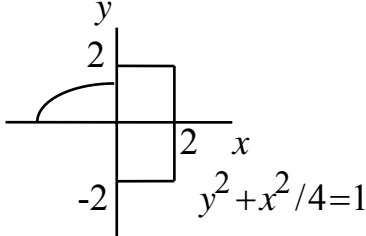
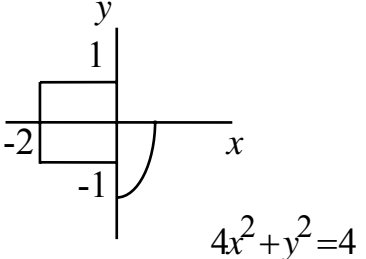
Продолжение табл. 2

№ п/п	$f_1(x, y)$	$f_2(x, y)$	Область
7	$2x + \operatorname{arctg}(y)$	$\operatorname{arctg}\left\{\frac{1}{x^2} + \frac{1}{y^2}\right\}$	
8	$\ln x^2 + 18y $	$\sqrt{\sin^2(x) + \cos^2(y)}$	
9	$x + \operatorname{tg}(y)$	$\ln x^3 + 3xy $	
10	$\operatorname{tg}\left(\frac{y}{\sin^2(x)}\right)$	$\sqrt{\frac{1}{x^2} + \frac{1}{y^2}}$	
11	$\frac{1}{\sin(x)} + \ln y $	$x^3 + x^2y + y^3$	
12	$\sqrt{x^2 + y^2}$	$x^2 - 10\sin(xy)$	

Продолжение табл. 2

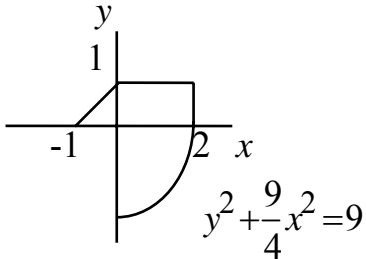
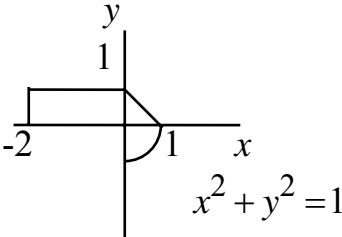
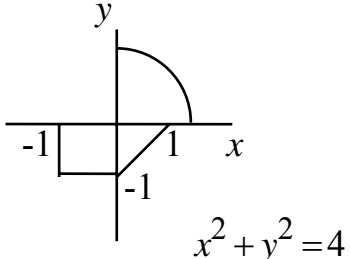
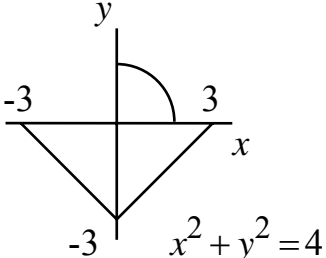
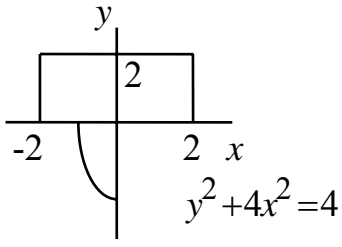
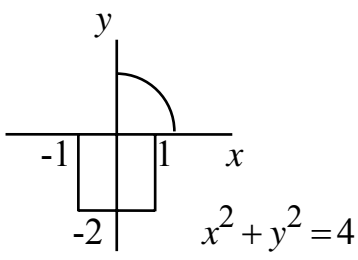
№ п/п	$f_1(x, y)$	$f_2(x, y)$	Область
13	$\sqrt{ x^2 + 10y }$	$\sqrt{x^2 + e^y}$	
14	$e^{\cos(y)+x}$	$\sqrt{ \sin(x) + \ln y  }$	
15	$2\sin(x)\cos(y)$	$\sqrt{ x^2 + xy + y^2 }$	
16	$\frac{\operatorname{tg}(x)}{e^y}$	$\sqrt[3]{\sin^2(2x+3y)}$	
17	$e^{5x} + 18\cos(y)$	$\frac{\sin(x^2 + y^2)}{x^2 + y^2}$	
18	$\ln\left(\frac{y}{\sin^2(x)}\right)$	$3\ln(x^2) + 5y$	

Продолжение табл. 2

№ п/п	$f_1(x, y)$	$f_2(x, y)$	Область
19	$(\sin(x) + \operatorname{tg}(y))^3$	$\sqrt{e^{1/x^2+1/y^2}}$	
20	$\frac{\operatorname{tg}(y)}{1 - \cos^2(x)}$	$\frac{\sin(x)}{\sqrt{ xy }}$	
21	$\sin^2(x) + \ln y $	$\frac{x + \sin(y)}{\cos(xy)}$	
22	$\ln y - 3\sin(x) $	$\operatorname{arctg}(e^x + y^2)$	
23	$\sqrt{ y } - \cos^2(x)$	$\operatorname{arctg}(e^y + x^2)$	
24	$\frac{\sin(x)\cos(y)}{1 - \sin^2(x)}$	$\frac{\sin^2(x) + \cos^2(y)}{\sqrt{x^2 + y^2}}$	



Окончание табл. 2

№ п/п	$f_1(x, y)$	$f_2(x, y)$	Область
25	$e^{x+\operatorname{tg}(y)}$	$\sqrt{e^{x+\sin(y)}}$	
26	$\cos\left(\frac{ x + y }{xy}\right)$	$\frac{\cos^2(x)}{\sin(2x+3y)}$	
27	$\ln \operatorname{arctg}(y/x) $	$\operatorname{arctg}\left(\frac{1}{x^2} + \frac{1}{y^2}\right)$	
28	$e^{(5x+8y)/xy}$	$\sqrt{ \ln (x) + y  }$	
29	$\sqrt{\operatorname{arctg}(y/x)}$	$\sqrt{\sqrt{x^2 + y^2} + y^2}$	
30	$\ln y  + e^{\sin(x)}$	$\sqrt{e^{x^2+y^2}}$	

## Задачи к защите лабораторной работы № 2

1. Даны числа  $a, b, c$ . Найти максимальное.
2. Даны числа  $a, b, c$ . Найти минимальное.
3. Даны числа  $a, b, c$ . Найти среднее.
4. Даны числа  $a, b, c$ . Выбрать из них те, которые принадлежат интервалу  $(d, l)$ .
5. Даны числа  $a, b, c$ . Каждое число заменить наибольшим, если числа образуют возрастающую последовательность. В остальных случаях числа оставить без изменений.
6. Даны числа  $a, b, c$ . Каждое число заменить наименьшим, если числа образуют убывающую последовательность. В остальных случаях числа оставить без изменений.
7. Даны числа  $a, b, c$ . Существует ли треугольник с длинами сторон  $a, b, c$  ?
8. Даны числа  $a, b, c$ . Выяснить, имеет ли квадратное уравнение  $ax^2 + bx + c = 0$  действительные корни. Если имеет, то найти их.
9. Даны числа  $a, b, c$ . Исследовать биквадратное уравнение  $ax^4 + bx^2 + c = 0$  на наличие действительных корней и найти их.
10. Выяснить, можно ли прямоугольник со сторонами  $a, b$  уместить внутри прямоугольника со сторонами  $c, d$ .
11. Дать ответ на вопрос: пройдет ли кирпич с ребрами  $a, b, c$  в прямоугольное отверстие со сторонами  $x, y$ .
12. Определить, является ли треугольник с координатами вершин  $(x_1, y_1), (x_2, y_2), (x_3, y_3)$  прямоугольным ?
13. Выяснить, является ли четырехугольник с координатами вершин  $(x_1, y_1), (x_2, y_2), (x_3, y_3), (x_4, y_4)$  квадратом.
14. Выяснить, является ли четырехугольник с координатами вершин  $(x_1, y_1), (x_2, y_2), (x_3, y_3), (x_4, y_4)$  ромбом.
15. Выяснить, является ли четырехугольник с координатами вершин  $(x_1, y_1), (x_2, y_2), (x_3, y_3), (x_4, y_4)$  прямоугольником.

16. Даны три числа  $x, y, z$ . Если их сумма меньше 1, то наименьшее заменить полусуммой двух других, иначе наименьшее заменить наибольшим.
17. Даны три числа  $x, y, z$ . Если их сумма больше 1, то наибольшее из  $x$  и  $y$  заменить полусуммой оставшихся, иначе наибольшее из трех чисел заменить наименьшим.
18. Можно ли построить четырехугольник со сторонами  $a, b, c, d$ ?
19. Для заданных значений  $a, b, c$  найти  $z = \max(\min(a, b), c)$ .
20. Даны четыре неравных числа  $a, b, c, d$ . Определить среднее значение двух чисел, которые не являются минимумом и максимумом.
21. Заданы три числа  $x, y, z$ . Если  $x < 0$ , то  $p$  задать как максимальное из  $y$  и  $z$ .
22. Заданы три числа  $x, y, z$ . Если  $z > 0$ , то  $p$  задать как минимальное из  $x$  и  $y$ .
23. Произвольный выпуклый четырехугольник задан координатами своих вершин на плоскости. Есть ли в этом четырехугольнике прямые углы?
24. Заданы три числа  $a, b, c$ . Если  $a < 0$ , то определить  $p$  как минимальное из  $b$  и  $c$ , иначе вычислить  $p$  как среднее арифметическое из  $b$  и  $c$ .
25. Заданы три числа  $a, b, c$ . Если  $a > 0$ , то  $p$  задать как максимальное из  $b$  и  $c$ , иначе вычислить  $p$  как среднее арифметическое из  $a, b$  и  $c$ .
26. Заданы три числа  $a, b, c$ . Если  $a = 0$ , то вычислить  $p$  как среднее арифметическое от  $b$  и  $c$ , иначе минимальное из трех чисел заменить максимальным.
27. Даны две окружности радиусом  $R_1$  и  $R_2$  с центром в начале координат. Определить, где находится точка с координатами  $(x, y)$ : внутри меньшей окружности, в кольце между окружностями или вне их.
28. Заданы координаты двух точек. Определить, лежат ли они на одной окружности с центром в начале координат.
29. Заданы площади круга  $S_1$  и квадрата  $S_2$ . Выяснить, поместится ли круг в квадрате.
30. Заданы площади круга  $S_1$  и квадрата  $S_2$ . Выяснить, поместится ли квадрат в круге.

## Лабораторная работа № 3

### ПРОГРАММИРОВАНИЕ ЦИКЛИЧЕСКИХ ВЫЧИСЛИТЕЛЬНЫХ ПРОЦЕССОВ

Цель работы. Получить навыки программирования циклических вычислительных процессов. Изучить циклы с шагом и циклы с условием.

#### Введение

Работа содержит задачи, которые сводятся к нахождению конечной суммы ряда  $S = \sum a_n(x)$  при различных значениях  $x$ . Каждое слагаемое  $a_n(x)$  зависит от величины  $x$  и номера  $n$ , определяющего место этого слагаемого в сумме. Ряды подобраны таким образом, чтобы при бесконечно большом числе членов ряда их сумма в указанном диапазоне изменения величины  $x$  была конечным числом, зависящим только от  $x$ , т.е. ряды являются сходящимися.

Например,  $S = \sum_{n=0}^{\infty} a_n(x) = \sum_{n=0}^{\infty} (-1)^n x^{2n+1} / (2n+1)! = \sin(x)$

Первый член этого ряда при  $n = 0$  равен  $x$ , второй член при  $n = 1$  равен  $(-x^3)/3!$ , третий член при  $n = 2$  равен  $x^5/5!$  и т.д. В сумме при бесконечно большом числе членов они дают функцию  $\sin(x)$ . Видно, что при ограниченных значениях  $x$ , сравнимых по модулю с единицей, каждый последующий член этого знакопеременного ряда существенно меньше предыдущего, поэтому ряд быстро сходится и для вычисления суммы ряда с необходимой точностью достаточно просуммировать относительно небольшое число членов ряда. Например, суммирование можно прекратить, когда текущий член ряда станет по абсолютной величине меньше заданного малого положительного числа  $\varepsilon$ , определяющего погрешность вычисления суммы ряда.

Формула общего члена суммы принадлежит к одному из следующих типов:

$$\text{а) } x^n/n! ; \quad (-1)^n x^{2n-1}/(2n-1)! ; \quad x^{2n}/(2n)!$$

$$\text{б) } \cos(nx)/n ; \quad \sin((2n-1)x)/(2n-1) ; \quad \cos(2nx)/(4n^2-1)$$

$$\text{в) } x^{4n+1}/(4n+1) ; \quad (-1)^n \cos(nx)/n^2 ; \quad (x/2)^2(n^2+1)/n!$$

В случае (а) для вычисления члена суммы целесообразно использовать рекуррентные соотношения, т.е. выражать последующий член суммы через предыдущий, например, если

$$a_n = x^n/n! , \text{ то } a_{n+1} = x^{n+1}/(n+1)! = \left(x^n/n!\right)(x/(n+1)) = a_n x/(n+1) .$$

В случае (б) каждый член суммы вычисляется согласно общей формуле, которую и нужно запрограммировать.

В случае (в) член суммы целесообразно представить в виде двух сомножителей, один из которых вычисляется по рекуррентному соотношению, а другой – непосредственно, например, если

$$a_n = x^{4n+1}/(4n+1) , \text{ то } a_n = b_n c_n , \text{ где } b_n = 1/(4n+1) , \quad c_n = x^4 c_{n+1} .$$

### Задание

1. Записать расчетные формулы для вычисления слагаемого и конечной суммы ряда  $S = \sum a_n(x)$ . Варианты заданий приведены в таблице 3.
2. Составить алгоритм вычисления с заданной точностью суммы конечного числа членов ряда в указанном диапазоне изменения параметра  $x$  с заданным шагом. Для получения шага диапазон изменения  $x$  разделить на 10.
3. Составить программу решения задачи. В программе для каждого значения  $x$  предусмотреть вычисление как приближенной, так и точной суммы соответствующего ряда  $y$  по приведенной в таблице 3 формуле.
4. Отладить и просчитать программу. Результат получить в виде таблицы, содержащей для каждого  $x$  из указанного диапазона приближенное и

точное значения суммы ряда, а также количество просуммированных членов.

### Пример выполнения задания.

Пусть требуется составить программу для вычисления суммы ряда:

$$S = -(2x)^2/2 + (2x)^4/24 + \dots + (-1)^n (2x)^{2n}/(2n)! + \dots$$

и функции  $y = 2(\cos^2(x) - 1)$  для  $0.1 \leq x \leq 1.0$ . Суммирование следует выполнять до тех пор, пока текущий член ряда по абсолютной величине не станет меньше заданного малого положительного числа  $\varepsilon$ .

Запишем ряд в виде

$$S = \sum a_n; \text{ где } a_n = (-1)^n (2x)^{2n}/(2n)! = c_n b_n$$

$$\text{Здесь: } c_n = (-1)^n; \quad b_n = (2x)^{2n}/(2n)!$$

Общий член ряда относится к типу (а), т.е. для вычисления члена суммы целесообразно использовать рекуррентные соотношения, выражая каждый последующий член суммы через предыдущий. Суммирование начинается с  $n = 1$ , при этом начальные значения  $c = -1$ ,  $b = 1$ .

Для каждого значения  $x$ , задаваемого во внешнем цикле, вычисление суммы ряда выполняется во внутреннем цикле, при прохождении которого номер члена ряда увеличивается на 1, а сумма изменяется по формуле  $S = S + a$ , где  $a$  вычисляется как произведение  $c$  и  $b$ . Начальное значение суммы в данном примере равно нулю  $S = 0$ . Суммирование в глухом цикле DO необходимо продолжать до тех пор, пока величина  $a$  не станет по абсолютной величине меньше  $\varepsilon$ . Вычисление постоянной по модулю знакопеременной величины  $c_n$  можно организовать в цикле по формуле  $c = -c$ . Для вычисления величины  $(2n)!$  в знаменателе необходимо во внутреннем цикле, где  $n$  каждый раз увеличивается на 1, число  $2n$  умножать на число  $(2n - 1)$ , так как факториал — это, по определению, произведение последовательных натуральных чисел.

## Фортран–программа

```

PROGRAM LAB3
REAL A, B, C, EPS, H, S, X, X0, XK, Y
INTEGER I, N
OPEN (1, FILE='LAB3.TXT')
X0=0.1; XK=1.; EPS=0.0001
H=(XK-X0)/10.
    DO I=0,10
        X=X0+H*I
        N=1; C=-1.; B=1.; S=0.
        DO
            B=B*(2.*X)**2/(2.*N*(2.*N-1.)); A=C*B
            N=N+1; S=S+A; C=-C;
            IF (ABS(A) < EPS) EXIT
        END DO
        Y=2.*(COS(X)**2-1.)
        PRINT 2, X,S,Y,N; WRITE (1,2) X,S,Y,N
    END DO
2  FORMAT (' X=',F6.3,3X,'S=',E10.3,3X,'Y=',E10.3,3X,'N=',I5)
END PROGRAM LAB3

```

Результаты счета выводятся не только на дисплей, но в файл LAB3.TXT. Вывод осуществляется под управлением оператора FORMAT с меткой 2, который формирует таблицу, содержащую для каждого  $x$  из указанного диапазона приближенное и точное значения суммы ряда, а также количество просуммированных членов.

**Таблица 3. Варианты заданий к лабораторной работе № 3**

№ п/п	Формула ряда	$x_0 \leq x \leq x_k$	Точное значение суммы ряда $y$
		$x_0$ $x_k$	

1	$S = 1 + x(\ln 3)/1 + x^2(\ln^2 3)/2! + \dots$ $+ x^n(\ln^n 3)/n! + \dots$	0.1	1.0	$3^x$
2	$S = \cos x + (\cos 2x)/2 + \dots$ $+ (\cos nx)/n + \dots$	$\pi/5$	$9\pi/5$	$-\ln(2 \sin(x/2))$
3	$S = x - x^3/3! + \dots$ $+ (-1)^n x^{2n+1}/(2n+1)! + \dots$	0.1	1.0	$\sin x$
4	$S = \sin x - (\sin 2x)/2 + \dots$ $+ (-1)^{n+1}(\sin nx)/n + \dots$	$\pi/5$	$4\pi/5$	$x/2$
5	$S = 1 + x/1 + x^2/2! + \dots + x^n/n! + \dots$	1.0	2.0	$e^x$
6	$S = 1 + x \cos(\pi/4) + \dots$ $+ x^n \cos(n\pi/4)/n! + \dots$	0.1	1.0	$\cos(x \sin(\pi/4))^*$ $e^{x \cos(\pi/4)}$
7	$= 1 - x^2/2! + \dots + (-1)^n x^{2n}/(2n)! + \dots$	0.1	1.0	$\cos x$
8	$S = x \sin(\pi/4) + \dots$ $+ x^n \sin(n\pi/4) + \dots$	0.1	0.8	$\frac{x \sin(\pi/4)}{1 - 2x \cos(\pi/4) + x^2}$
9	$S = x + x^5/5 + \dots$ $+ x^{4n+1}/(4n+1) + \dots$	0.1	0.8	$\frac{1}{4} \ln \frac{1+x}{1-x} + \frac{1}{2} \operatorname{arctg} x$
10	$S = 1 + \cos x + \dots + \cos(nx)/n! + \dots$	0.1	1.0	$\cos(\sin x) e^{\cos x}$
11	$S = 1 + 3x^2 + \dots + (2n+1)x^{2n}/n! + \dots$	0.1	1.0	$(1 + 2x^2)e^{x^2}$
12	$S = x \cos(\pi/3) + x^2 \cos(2\pi/3)/2 + \dots$ $+ x^n \cos(n\pi/3)/n + \dots$	0.1	0.8	$\frac{-\ln(x^2 - 2x \cos(\pi/3) + 1)}{2}$



Продолжение табл. 3

№ п/п	Формула ряда	$x0 \leq x \leq xk$		Точное значение суммы ряда у
		$x0$	$xk$	
13	$S = \frac{x-1}{x+1} + \frac{1}{3} \left( \frac{x-1}{x+1} \right)^3 + \dots$ $+ \frac{1}{2n+1} \left( \frac{x-1}{x+1} \right)^{2n+1} + \dots$	0.2	1.0	$\frac{\ln x}{2}$
14	$S = -\cos x + \frac{\cos 2x}{4} + \dots$ $+ (-1)^n \frac{\cos nx}{n^2} + \dots$	$\pi/5$	$\pi$	$\frac{1}{4} (x^2 - \pi^2/3)$
15	$S = x^3/3 - x^5/15 + \dots$ $+ (-1)^{n+1} x^{2n+1} / (4n^2 - 1) + \dots$	0.1	1.0	$\frac{1+x^2}{2} \operatorname{arctg} x - \frac{x}{2}$
16	$S = \sin x + \frac{1}{3} \sin 3x + \dots$ $+ \frac{1}{2n-1} \sin((2n-1)x) + \dots$	$\pi/10$	$9\pi/10$	$\pi/4$
17	$S = 1 + x^2/2! + \dots + x^{2n}/(2n)! + \dots$	0.1	1.0	$\frac{1}{2} (e^x + e^{-x})$
18	$S = \frac{1}{3} \cos 2x + \frac{1}{15} \cos 4x + \dots$ $+ \frac{1}{4n^2 - 1} \cos(2nx) + \dots$	0.1	0.8	$\frac{1}{2} - \frac{\pi}{4}  \sin x $
19	$S = 1 + 2x + \dots + (2x)^n/n! + \dots$	0.1	1.0	$e^{2x}$
20	$S = 1 + 2\frac{x}{2} + \dots + \frac{n^2+1}{n!} \left( \frac{x}{2} \right)^n + \dots$	0.1	1.0	$\left( 1 + x/2 + x^2/4 \right) e^{x/2}$
21	$S = x - x^3/3 + \dots + (-1)^n \frac{x^{2n+1}}{2n+1} + \dots$	0.1	0.5	$\operatorname{arctg} x$

Окончание табл. 3

№ п/п	Формула ряда	$x0 \leq x \leq xk$		Точное значение суммы ряда $y$
		$x0$	$xk$	
22	$S = 1 - \frac{3}{2}x^2 + \dots$ $+(-1)^n \frac{2n^2 + 1}{(2n)!} x^{2n} + \dots$	0.1	1.0	$\left(1 - \frac{x^2}{2}\right) \cos x - \frac{x}{2} \sin x$
23	$S = -\frac{(2x)^2}{2} + \frac{(2x)^4}{24} + \dots$ $+(-1)^n \frac{(2x)^{2n}}{(2n)!} + \dots$	0.1	1.0	$2(\cos^2(x) - 1)$
24	$S = -(1+x)^2 + \frac{(1+x)^4}{2} + \dots$ $+(-1)^n \frac{(1+x)^{2n}}{n} + \dots$	-2.0	-1.0	$\ln \frac{1}{2 + 2x + x^2}$
25	$S = x + \frac{x^3}{3!} + \dots + \frac{x^{2n+1}}{(2n+1)!} + \dots$	0.1	1.0	$\frac{1}{2}(e^x - e^{-x})$
26	$S = x/3! + 4x^2/5! + \dots$ $+n^2 x^n / (2n+1)! + \dots$	0.2	0.8	$\frac{1}{4} \left( \frac{x+1}{\sqrt{x}} \operatorname{sh}(\sqrt{x}) - \operatorname{ch}(\sqrt{x}) \right)$
27	$S = x \cos(\pi/4) + x^2 \cos(2\pi/4) + \dots$ $+x^n \cos(n\pi/4) + \dots$	0.1	0.8	$\frac{x \cos(\pi/4) - x^2}{1 - 2x \cos(\pi/4) + x^2}$
28	$S = 3x + 8x^2 + \dots + n(n+2)x^n + \dots$	0.1	0.8	$x(3-x)/(1-x)^3$
29	$S = \cos x + \frac{\cos(3x)}{9} + \dots$ $+ \frac{\cos((2n-1)x)}{(2n-1)^2} + \dots$	$\pi/5$	$\pi$	$\pi^2/8 - (\pi/4) x $
30	$S = x^2/2 - x^4/12 + \dots$ $+(-1)^{n+1} \frac{x^{2n}}{2n(2n-1)} + \dots$	0.1	0.8	$x \operatorname{arctg} x - \ln \sqrt{1+x^2}$

### Задачи к защите лабораторной работы №3

1. Получить таблицу температур по Цельсию от 0 до 100 градусов с шагом 10 градусов и эквивалентных температур по шкале Фаренгейта, используя для перевода формулу  $T_f = 32 + \frac{9}{5}T_c$ .
2. Вычислить последовательности значений функций  $P_1(x) = x$ ,  $P_2(x) = (3x^2 - 1)/2$ ,  $P_3(x) = (5x^2 - 3x)/2$  для значений аргумента  $x = 1, 2, \dots, 20$ .
3. На отрезке  $[a, b]$  с шагом  $h = (b - a)/n$ , где  $n$  – натуральное число, вычислить значения функции  $y = \sin x + \cos 2x$  при  $a = -\pi$ ,  $b = \pi$ ,  $n = 20$ .
4. На отрезке  $[a, b]$  с шагом  $h = (b - a)/n$ , где  $n$  – натуральное число, вычислить значения функции  $y = \sin \sqrt{(2x + \cos x)}$  при  $a = 0$ ,  $b = 2\pi$ ,  $n = 15$ .
5. На отрезке  $[a, b]$  с шагом  $h = (b - a)/n$ , где  $n$  – натуральное число, вычислить значения функции  $y = |x|e^{-x}$  при  $a = -1$ ,  $b = 3$ ,  $n = 20$ .
6. Даны натуральные числа  $m$  и  $n$ . Получить  $\frac{m! + n!}{(m + n)!}$
7. Вычислить  $\prod_{i=1}^{40} \frac{i^2}{i^2 + 2i + 3}$
8. Вычислить  $\prod_{i=1}^{10} \left(2 + \frac{1}{i!}\right)$
9. Вычислить  $\prod_{i=1}^{50} \frac{i+1}{i+2}$
10. Вычислить  $\prod_{i=1}^{10} \left(1 - \frac{1}{i!}\right)^2$
11. Даны натуральное число  $n$  и действительное  $x$ . Вычислить  $\prod_{k=1}^n \left(1 + \frac{\sin kx}{k!}\right)$

12. Даны натуральное число  $n$  и действительное  $x$ . Вычислить

$$\prod_{k=1}^n \left( \frac{k}{k+1} - \cos^k(x) \right)$$

13. Дано действительное число  $a > 0$ . Последовательность  $x_1, x_2, \dots$  образована

по закону  $x_n = \frac{4}{5}x_{n-1} + \frac{a}{5}x_{n-1}$ , где  $x_0 = \min(2a, 5)$ . Найти значение  $n$ , для

которого  $|x_n - x_{n-1}| < 10^{-5}$ . Вычислить разность  $a - x_n^3$ .

14. Даны действительные положительные числа  $a, x$ . В последовательности  $y_1,$

$y_2, \dots$ , образованной по закону  $y_i = \frac{1}{2}(y_{i-1} + x/y_{i-1})$ , где  $y_0 = a$ , найти

первый член  $y_n$ , для которого  $|y_n^2 - y_{n-1}^2| < 10^{-4}$ .

15. Пусть  $x_0 = 1$ ,  $x_k = (2 - x_{k-1}^3)/5$ , где  $k = 1, 2, \dots$ . Найти  $n$ , для которого

$$|x_n - x_{n-1}| < 10^{-5}.$$

16. Пусть  $a_0 = a_1 = 1$ ,  $a_i = a_{i-2} + a_{i-1}/2^{i-1}$ , где  $i = 2, 3, \dots$ . Найти произведение

$$a_0 * a_1 * \dots * a_9.$$

17. Дано натуральное число  $n$ . Сколько цифр в числе  $n$ ?

18. Дано натуральное число  $n$ . Чему равна сумма его цифр?

19. Дано натуральное число  $n$ . Найти первую цифру числа  $n$ .

20. Дано натуральное число  $n$ . Получить сумму двух последних цифр числа  $n$ .

21. Дано натуральное число  $n$ . Выяснить, входит ли цифра 3 в запись числа  $n$ .

22. Дано натуральное число  $n$ . Переставить первую и последнюю цифры числа.

23. Дано натуральное число  $n$ . Вычислить

$$1/\sin 1 + 1/(\sin 1 + \sin 2) + \dots + 1/(\sin 1 + \dots + \sin n).$$

24. Дано натуральное число  $n$ . Вычислить

$$\cos 1 / \sin 1 + (\cos 1 + \cos 2) / (\sin 1 + \sin 2) + \dots + (\cos 1 + \dots + \cos n) / (\sin 1 + \dots + \sin n).$$

25. Дано действительное число  $a$  и натуральное  $n$ . Вычислить

$$a(a+1) * \dots * (a+n-1).$$

26. Дано действительное число  $a$  и натуральное  $n$ . Вычислить

$$\frac{1}{a} + \frac{1}{a(a+1)} + \dots + \frac{1}{a(a+1)\dots(a+n)}.$$

27. Дано действительное число  $a$  и натуральное  $n$ . Вычислить

$$a(a-n)(a-2n)\dots(a-n^2).$$

28. Дано действительное число  $a$  и натуральное  $n$ . Вычислить

$$\sin x + \sin \sin x + \dots + \underbrace{\sin \sin \dots \sin x}_n.$$

29. Дано действительное число  $a$ . Найти среди чисел  $1, 1+1/2, 1+1/2+1/3, \dots$  первое, большее  $a$ .

30. Дано действительное число  $a$ . Найти такое наименьшее  $n$ , что

$$1+1/2+\dots+1/n > a.$$

## Лабораторная работа № 4

### ИСПОЛЬЗОВАНИЕ МАССИВОВ

Цель работы. Изучить особенности описания и использования массивов в программе.

Задание. Разработать алгоритм решения выбранной по номеру варианта задачи, составить, отладить и просчитать программу с использованием массивов.

#### Введение

Наряду с простыми переменными в алгоритмических языках используются переменные с индексами, называемые массивами. Массив – это упорядоченная последовательность величин, обозначаемая одним именем. Величины, из которых состоит массив, называются элементами массива. Элемент массива обозначается с помощью имени массива и индексов, следующих в круглых скобках за именем массива. Индексы отделяются друг от друга запятыми и указывают положение элемента в массиве. Индексов может быть 1, 2 и т.д. вплоть до 7. Соответственно массивы бывают одномерные, двумерные и т.д.

Все массивы, используемые в программе, должны быть описаны, так как для хранения элементов массива в определенной последовательности отводится необходимый объем памяти. При описании указываются имя массива и затем в круглых скобках через запятую граничные пары, т.е. разделенные двоеточием минимальные и максимальные значения соответствующего индекса. При отсутствии двоеточия минимальное значение соответствующего индекса в Фортране по умолчанию равно единице. Для описания массивов в Фортране используются операторы описания типа переменных (REAL, INTEGER, COMPLEX, LOGICAL, CHARACTER, DOUBLE PRECISION), а также атрибут и оператор DIMENSION.

Рассмотрим несколько примеров описания массивов. Значения граничных пар можно задавать с использованием именованных констант и константных выражений.

```
INTEGER, PARAMETER :: N=4
```

```
REAL, DIMENSION :: A(N,N), B(N), K(N+1,N+1), C(0:N)
```

В этом примере оператор REAL с атрибутом DIMENSION описывает двумерный вещественный массив A, содержащий 16 элементов; одномерный вещественный массив B, содержащий 4 элемента; двумерный вещественный массив K, содержащий 25 элементов, и одномерный вещественный массив C, содержащий 5 элементов.

Объекты с общими атрибутами можно группировать в единый список, например, следующий оператор описывает два двумерных вещественных массива D и E, каждый из которых имеет 3 строки и 4 столбца.

```
REAL, DIMENSION (3,4) :: D, E
```

Операторы описания типа могут использоваться для описания массивов и без атрибутов, но с указанием граничных пар, например,

```
INTEGER MIN(2,2), MAX(10)
```

```
REAL Q(5), SUMMA(3,3)
```

Здесь: MIN и MAX – целые массивы, содержащие 4 и 10 элементов соответственно; Q и SUMMA – вещественные массивы, содержащие 5 и 9 элементов.

Оператор DIMENSION также может использоваться для описания массивов, например, для описания одномерного вещественного массива F из 10 элементов

```
REAL F
```

```
DIMENSION F(-2:7)
```

Использование оператора DIMENSION без операторов описания типа, когда тип массива определяется по умолчанию возможно, но нежелательно.

Операторы, описывающие массивы, необходимо располагать в начале программы перед первым исполняемым оператором.

В Фортране список операторов ввода READ и вывода PRINT или WRITE может содержать как имена простых переменных и элементов массива, так и имена массивов. В последнем случае вводу-выводу подлежат все элементы массива в «естественной последовательности» (т.е. сначала растет первый индекс, затем второй и т.д.). Так, если в программе описан массив A(2,2) и имеется оператор ввода READ \*, A, то численные значения необходимо задавать в следующей последовательности: A(1,1), A(2,1), A(1,2), A(2,2), т.е. любая матрица вводится по столбцам. Для изменения порядка ввода-вывода или если требуется передать только часть элементов массива можно использовать неявную форму оператора цикла. Например, ввод матрицы A(2,2) по строкам осуществляет оператор READ \*, ((A(I,J), J=1,2), I=1,2).

Не следует забывать, что в Фортране функции «что вводить (выводить)» и «как вводить (выводить)» распределены между операторами ввода-вывода и оператором FORMAT.

Если в программе предусмотрен ввод исходных данных с клавиатуры, то в процессе отладки приходится задавать одни и те же численные значения много раз. Поэтому, когда исходных данных достаточно много, удобнее записать их перед счетом программы в файл данных, а в программе предусмотреть оператор, считывающий данные из файла данных. Кроме того, результаты также имеет смысл выводить в файл данных, чтобы иметь возможность получить распечатку результатов счета. Обращение к файлу данных производится по его логическому номеру N, указанному в операторах READ (N,...) или WRITE (N,...). С помощью оператора OPEN можно как назначать логические номера уже имеющимся файлам, так и создавать новые файлы с указанием соответствующего логического номера, например, OPEN (1, FILE='LAB4.TXT'). После выполнения этого оператора файлу LAB4.TXT будет соответствовать логический номер 1. Файл LAB4.TXT создается в папке проекта либо предварительно до запуска программы на счет, либо в процессе работы программы в момент выполнения оператора OPEN. Для завершения доступа к файлу используется оператор CLOSE.

### **Типовые примеры использования массивов в программах.**



Отличительной особенностью Фортрана является развитая система операций с массивами. Операции над массивами и сечениями массивов задают параллелизм действий над компонентами массивов (или массива). Такие средства, с одной стороны, позволяют пользователю лаконично и сжато описать алгоритм обработки массивов и, с другой стороны, дают возможность компилятору генерировать более эффективный код с учетом особенностей конкретного компьютера. Очевидно, что наиболее эффективно использование этих возможностей для вычислительных систем, имеющих аппаратные средства для векторной обработки.

В Фортран встроено большое число функций, позволяющих:

- выполнять вычисления в массивах, например, суммировать элементы массива или находить его максимальный элемент;
- преобразовывать массивы, например, можно получать из одномерного массива двумерный;
- получать справочные данные о массиве (размер, форма и значения границ каждого измерения).

1. Ввод одномерного массива B(8) можно осуществить несколькими способами:

а) `READ (1,*) B`

Численные значения (8 значений) перед запуском программы на счет записываются через запятую или пробел в файл данных, соответствующий логическому номеру 1. Этот способ наиболее предпочтителен.

б) `PRINT 1`

`1     FORMAT (' INPUT VECTOR B(8)')`

`READ *, B`

После появления на экране монитора текста `INPUT VECTOR B(8)` численные значения элементов массива вводятся с клавиатуры (8 значений через запятую или пробел).

в) `PRINT 1`

```

1      FORMAT (' INPUT VECTOR B')
      DO I=1,8
        PRINT 2, I
        READ *, B(I)
      END DO
2      FORMAT (' B(' ,I2,')=')

```

Массив вводится с клавиатуры поэлементно по одному значению в строку после появления на экране соответствующей подсказки.

## 2. Ввод двумерного массива A(4,4)

а) из файла данных построчно с использованием неявной формы цикла

```
READ (1,*) ((A(I,J), J=1,4), I=1,4)
```

Перед выполнением программы численные значения записываются в файл данных, соответствующий логическому номеру 1, в четыре строки по четыре элемента в строку через запятую или пробел.

б) из файла данных построчно с использованием сечения массива (простейшей формы индексного триплета)

```
READ (1,*) (A(I,:),I=1,4)
```

Файл данных такой же, как и в случае а).

в) с клавиатуры построчно через запятую или пробел

```
READ *, (A(I,:),I=1,4)
```

г) с клавиатуры построчно (с подсказкой о номере строки)

```

      DO I=1,4
        PRINT 2, I
        READ *, (A(I,J),J=1,4)
      END DO
2      FORMAT (' INPUT',I2,' STROKI')

```

## 3. Вывод одномерного массива B(4) на экран.

```
PRINT 5, B
```

5     FORMAT (' VECTOR B'/4F10.4)

При выводе в файл данных оператор PRINT заменяется на оператор WRITE, например, при выводе двумерного массива A(4,4)

WRITE (1,10) (A(I,:), I=1,4)

10    FORMAT (' MATRICA A'/4(4F10.4/))

4. Суммирование элементов одномерного массива B(10)  $S = \sum_{i=1}^{10} b_i$

а) с использованием встроенной функции SUM

S=SUM(B)

б) по следующему алгоритму: начальное значение суммы принимается равным нулю  $S = 0$ , затем в цикле по  $i$  от 1 до 10 накапливается сумма  $S = S + b_i$

S=0.

DO I=1,10

S=S+B(I)

END DO

5. Произведение элементов одномерного массива B(10)  $P = \prod_{i=1}^{10} b_i$

а) целесообразно использовать встроенную функцию PRODUCT

P=PRODUCT(B)

б) можно составить алгоритм, аналогичный алгоритму сложения элементов массива, в котором начальное значение произведения необходимо положить равным единице, а операцию сложения заменить произведением

P=1.

DO I=1,10

P=P\*B(I)

END DO

6. Пусть требуется определить максимальный элемент массива B из N элементов и его номер K.

а) Определение максимального (или минимального) элемента одномерного массива  $S = \max(b_1, b_2, \dots, b_n)$  и номера  $k$  этого элемента выполняется по следующему алгоритму: сначала за максимальный принимается первый элемент массива  $S = b_1$  и, следовательно, номером  $k$  максимального элемента будет единица  $k = 1$ . Затем в цикле по  $i$  от 2 до  $N$  текущее значение максимального  $S$  сравнивается с  $b_i$ . Если условие  $S < b_i$  не выполняется, то цикл повторяется. Если условие  $S < b_i$  выполняется, то текущему значению максимального  $S$  присваивается значение  $b_i$ , а его номеру – значение  $i$ .

```

INTEGER, PARAMETER :: N=10
REAL B(N) /1.,2.,3.,4.,5.,6.,5.,4.,3.,2./
S=B(1); K=1
DO I=2,N
    IF (S.LT.B(I)) THEN
        S=B(I); K=I
    END IF
END DO
PRINT *, ' S=', S, ' K=', K
END

```

б) Определение максимального (или минимального) элемента одномерного массива  $S = \max(b_1, b_2, \dots, b_n)$  и номера  $k$  этого элемента можно выполнить с использованием встроенных функций MAXVAL и MAXLOC.

```

INTEGER, PARAMETER :: N=10
REAL B(N) /1.,2.,3.,4.,5.,6.,5.,4.,3.,2./
S=MAXVAL(B); K=MAXLOC(B)
PRINT *, ' S=', S, ' K=', K
END

```

7. Пусть требуется расположить элементы одномерного массива  $B(10)$  в порядке невозрастания, т.е. так чтобы для всех элементов массива выполнялось условие:  $b_i \geq b_{i+1}$  при  $i = 1, 9$ . Рассмотрим один из способов решения этой задачи.

Сначала из всех элементов  $b_i$  при  $i = 1, 10$  найти максимальный (его величину  $S$  и номер  $k$ ), затем переставить первый и максимальный элементы. Из оставшихся девяти элементов  $b_i$  при  $i = 2, 10$  найти максимальный и поменять его местами со вторым и т.д. Перестановка двух элементов массива с номерами  $k$  и  $j$  выполняется следующим образом: элемент  $b_k$  равен  $S$ , поэтому сначала следует заменить  $b_k$  на  $b_j$ , а затем  $b_j$  присваивается значение  $S$ . Программа упорядочения элементов массива будет иметь следующий вид:

```

DO J=1,9
  S=B(J); K=J
  DO I=J+1,10
    IF (S.LT.B(I)) THEN
      S=B(I); K=I
    END IF
  END DO
  B(K)=B(J); B(J)=S
END DO

```

Внешний цикл по  $j$  изменяет номер элемента, с которого начинается поиск.

8. Пусть требуется составить одномерный массив  $C(4)$  из максимальных элементов строк матрицы  $A(4,4)$ . Программа строится по схеме двойного цикла. Параметром внешнего цикла является номер строки  $i$ . Во внутреннем цикле по  $j$  определяется максимальный элемент строки:

```

DO I=1,4
  C(I)=A(I,1)
  DO J=2,4
    IF (C(I).LT.A(I,J)) C(I)=A(I,J)
  END DO
END DO

```

9. Пусть требуется умножить матрицу  $A(4,4)$  на вектор  $B(4)$ . Результатом умножения матрицы  $A(4,4)$  на вектор  $B(4)$  является вектор  $C(4)$ , элементы

которого определяются по формуле  $C_i = \sum_{j=1}^4 a_{ij}b_j$ .

а) используем встроенную функцию MATMUL, которая выполняет матричное умножение операндов

```
C=MATMUL(A,B)
```

б) алгоритм строится по схеме двойного цикла: внешний цикл по параметру  $i$ , во внутреннем цикле по параметру  $j$  производится суммирование:

```
DO I=1,4
  C(I)=0.
  DO J=1,4
    C(I)=C(I)+A(I,J)*B(J)
  END DO
END DO
```

10. Пусть требуется найти произведение  $D$  матрицы  $A$  на матрицу  $F$ , где  $A(4,4)$ ,  $F(4,4)$  и  $D(4,4)$ . Умножение матрицы на матрицу осуществляется по схеме

тройного цикла в соответствии с формулой  $d_{ij} = \sum_{k=1}^4 a_{ik}f_{kj}$ , где  $i = 1,4$  и  $j = 1,4$ .

а) используем схему тройного цикла с суммированием во внутреннем цикле

```
DO I=1,4
  DO J=1,4
    D(I,J)=0.
    DO K=1,4
      D(I,J)=D(I,J)+A(I,K)*F(K,J)
    END DO
  END DO
END DO
```

б) используем схему двойного цикла со встроенной функцией суммирования

```
DO I=1,4
  DO J=1,4
    D(I,J)=SUM(A(I,:)*F(:,J))
  END DO
END DO
```

11. Пусть требуется вычислить элементы матрицы D(4,4) по формулам:

$$d_{ij} = \left\{ \begin{array}{ll} (b_i + b_j)^2, & b_i + b_j < -1 \\ 2(b_i + b_j), & -1 \leq b_i + b_j \leq 1 \\ \sqrt{b_i + b_j}, & b_i + b_j > 1 \end{array} \right\}$$

Программа, использующая блочный оператор IF, имеет следующий вид

```
DO I=1,4
  DO J=1,4
    IF ((B(I)+B(J)).LT.-1.) THEN
      D(I,J)=(B(I)+B(J))**2
    ELSE IF ((B(I)+B(J)).GT.1.) THEN
      D(I,J)=SQRT(B(I)+B(J))
    ELSE
      D(I,J)=2.*(B(I)+B(J))
    END IF
  END DO
END DO
```

#### **Варианты заданий к лабораторной работе № 4**

В вариантах заданий в качестве исходных данных используются следующие массивы:

$$A = \begin{pmatrix} -1.06 & 3.29 & -2.93 & 0.46 \\ 1.85 & -4.05 & 1.96 & -2.17 \\ 0.36 & 2.85 & -3.02 & 1.57 \\ 2.87 & -1.83 & 2.89 & -0.93 \end{pmatrix}; \quad B = \begin{pmatrix} 1.07 \\ -2.36 \\ 1.98 \\ 0.76 \end{pmatrix}.$$

1. Заданы массивы A(4,4), B(4). Вычислить элементы массивов C(4), D(4,4), F(4,4) в соответствии с формулами:

$$c_i = \begin{cases} \operatorname{tg}(0.1i + 1), & i = 1, 2 \\ \operatorname{ctg}(0.2i + 1), & i = 3, 4 \end{cases} \quad d_{ij} = c_i + b_j, \quad f_{ij} = a_{ij} + d_{ij}$$

Вывести на печать массивы C, D, F

2. Заданы массивы A(4,4), B(4). Вычислить элементы массива по формуле:

$$c_i = \begin{cases} \sqrt{b_i}, & b_i > 0 \\ b_i^2, & b_i \leq 0 \end{cases}$$

Найти максимальный элемент массива P=C+B. Преобразовать матрицу A, разделив все ее элементы на найденный максимум. Печатать A, B, C, P.

3. Заданы массивы A(4,4), B(4). Из 1-й строки A вычесть 3-ю и в полученной матрице найти максимальный элемент. Из всех элементов массива B вычесть найденный максимум. Печатать A, B.

4. Задан массив B(4). Вычислить элементы массива C(4) по формулам

$$c_i = \begin{cases} b_i^2, & b_i < -1 \\ 1 + b_i, & -1 \leq b_i \leq 1 \\ \sqrt{b_i}, & b_i > 1 \end{cases}$$

Вычислить элементы матрицы D(4,4) по формуле  $d_{ij} = b_i c_j$ . Найти  $S = \sum_{i=1}^4 b_i c_i$

Из всех элементов массива B вычесть S. Печатать B, C, D, S.

5. Задан массив A(4,4). Составить вектор C(4) из минимальных элементов в столбцах A. Из каждой строки A вычесть C. Печатать A, C.
6. Задан массив A(4,4). Составить вектор C(4) из максимальных модулей элементов в строках A. Элементы вектора C разделить на максимальный элемент C. Из диагональных элементов A вычесть C. Печатать A, C.



7. Заданы массивы  $A(4,4)$ ,  $B(4)$ . Найти  $C=AB$ . Максимальный элемент  $C$  и его номер  $k$ . Из  $k$ -й строки матрицы  $A$  вычесть  $C$ . Печатать  $A$ ,  $B$ ,  $C$ ,  $k$ .

8. Заданы массивы  $A(4,4)$ ,  $B(4)$ . Найти максимальные элементы массивов  $A$  и  $B$  и поменять их местами. Печатать  $A$ ,  $B$ .

9. Задан массив  $B(4)$ . Вычислить элементы массива  $C(4)$  по формулам

$$c_i = \begin{cases} 1 + \sqrt{b_i}, & b_i > 0 \\ 2b_i^2, & b_i \leq 0 \end{cases}$$

Вычислить элементы матриц  $D(4,4)$  ( $d_{ij} = b_i + c_j$ ) и  $F(4,4)$  ( $f_{ij} = b_i c_j$ ).

Найти  $Q=DF$ . Печатать  $C$ ,  $D$ ,  $F$ ,  $Q$ .

10. Заданы массивы  $A(4,4)$ ,  $B(4)$ . Составить вектор  $C(4)$  из максимальных элементов в строках  $A$ . Вычислить  $P=BC$ . Из диагональных элементов  $A$  вычесть  $P$ . Печатать  $A$ ,  $B$ ,  $C$ ,  $P$ .

11. Задан массив  $A(4,4)$ . Вычислить элементы массива по формулам

$$d_{ij} = \begin{cases} \sqrt{a_{ij}}, & a_{ij} > 0 \\ a_{ij}^2 - 1, & a_{ij} \leq 0 \end{cases}$$

Найти  $F=AD$ . Печатать  $A$ ,  $D$ ,  $F$ .

12. Заданы массивы  $A(4,4)$ ,  $B(4)$ . Найти максимальный элемент  $B$  и его номер  $k$ , минимальный элемент  $B$  и его номер  $l$ . В матрице  $A$  элемент  $a_{kl}$  заменить на найденный максимум,  $a_{lk}$  – на минимум. В векторе  $B$  поменять местами максимальный и минимальный элементы. Вычислить  $C=AB$ . Печатать  $A$ ,  $B$ ,  $C$ .

13. Заданы массивы  $A(4,4)$ ,  $B(4)$ . Упорядочить  $B$  в порядке не убывания. Из каждой строки  $A$  вычесть  $B$ . Печатать  $A$ ,  $B$ .

14. Заданы массивы  $A(4,4)$ ,  $B(4)$ . Вычислить  $C=AB$ . Упорядочить  $C$  в порядке не возрастания. Из столбцов  $A$  вычесть  $C$ . Печатать  $A$ ,  $B$ ,  $C$ .

15. Задан массив  $B(4)$ . Вычислить  $C(4)$  по формулам:

$$c_1 = \max(b_i), \quad c_2 = \min(b_i), \quad c_3 = c_1 + c_2, \quad c_4 = c_1 - c_2$$

Вычислить элементы  $D(4,4)$  по формулам  $d_{ij} = \begin{cases} b_i - c_j, & b_i > 0 \\ b_i + c_j, & b_i \leq 0 \end{cases}$

Печатать  $B, C, D$ .

16. Заданы массивы  $A(4,4), B(4)$ . Упорядочить строки  $A$  по не убыванию значений первых элементов строк. Вычислить элементы вектора  $C(4)$  по формуле  $c_i = a_{ii} - b_i$ . Печатать  $A, B, C$ .

17. Задан массив  $B(4)$ . Вычислить элементы матрицы  $D(4,4)$  по формулам

$$d_{ij} = \begin{cases} \sin(b_i + b_j), & 0 \leq |b_i + b_j| < \pi \\ \cos(b_i + b_j), & \pi \leq |b_i + b_j| < 2\pi \\ \sin(b_i - b_j), & 2\pi \leq |b_i + b_j| \end{cases}$$

Найти максимальные элементы  $D$  и  $B$ , поменять их местами. Печатать  $B, D$ .

18. Задан массив  $B(4)$ . Вычислить элементы матрицы  $D(4,4)$  по формулам

$$d_{ij} = \begin{cases} 2(b_i + b_j)^2, & b_i + b_j < -2 \\ 4(b_i + b_j), & -2 \leq b_i + b_j \leq 2 \\ \sqrt{b_i + b_j}, & b_i + b_j > 2 \end{cases}$$

Из всех элементов  $D$  вычесть максимальный элемент  $B$ . Вычислить  $C=DB$ .

Печатать  $B, C, D$ .

19. Задан массив  $A(4,4)$ . Вычислить элементы вектора  $C(4)$  по формулам:

$$c_1 = \max(a_{ij}), \quad c_2 = (\max(a_{ij}) + \min(a_{ij}))/2,$$

$$c_3 = (\max(a_{ij}) - \min(a_{ij}))/2, \quad c_4 = \min(a_{ij})$$

Вычислить  $P=AC$ . Из диагональных элементов  $A$  вычесть вектор  $P$ .

Печатать  $A, C, P$ .

20. Заданы массивы  $A(4,4), B(4)$ . В матрице  $A$  найти минимальный и максимальный элементы, поменять их местами. Вычислить  $C=AB$ . Печатать  $A, B, C$ .

21. Заданы массивы  $A(4,4)$ ,  $B(4)$ . В массиве  $B$  найти максимальный и минимальный элементы, поменять их местами. Вычислить  $C=AB$ . Из диагональных элементов  $A$  вычесть вектор  $C$ . Печатать  $A$ ,  $B$ ,  $C$ .
22. Задан массив  $A(4,4)$ . Из диагональных элементов  $A$  вычесть максимальный элемент  $A$ . Вычислить вектор  $C(4)$  как сумму 1-й и 3-й строк  $A$ . Вычислить  $P=AC$ . Печатать  $A$ ,  $C$ ,  $P$ .
23. Заданы массивы  $A(4,4)$ ,  $B(4)$ . Вычислить  $C=AB$ . Упорядочить  $C$  в порядке не убывания. Из диагональных элементов  $A$  вычесть вектор  $C$ . Печатать  $A$ ,  $B$ ,  $C$ .
24. Заданы массивы  $A(4,4)$ ,  $B(4)$ . Вычислить элементы массива  $C(4)$  как сумму элементов в соответствующих строках  $A$ . Из диагональных элементов  $A$  вычесть  $C$ . Вычислить  $P=AC$ . Печатать  $A$ ,  $C$ ,  $P$ .
25. Задан массив  $A(4,4)$ . Вычислить элементы массива  $D(4,4)$  по формулам

$$d_{ij} = \begin{cases} a_{ij}/2, & i < j \\ 2a_{ij}^2, & i \geq j \end{cases}$$

Вычислить  $F=AD$ . Печатать  $A$ ,  $D$ ,  $F$ .

26. Задан массив  $B(4)$ . Вычислить элементы массива  $C(4)$  по формуле  $c_k = \sum_{i=1}^k b_i$

Вычислить элементы массива  $D(4,4)$  по формуле  $d_{ij} = b_i c_j$ . Печатать  $B$ ,  $C$ ,  $D$ .

27. Задан массив  $B(4)$ . Вычислить элементы массива  $C(4)$  по формуле  $c_k = \prod_{i=1}^k b_i$

Вычислить элементы массива  $D(4,4)$  по формуле  $d_{ij} = (b_i + c_j)^2$

Печатать  $B$ ,  $C$ ,  $D$ .

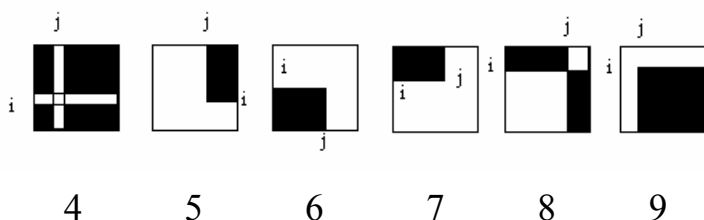
28. Заданы массивы  $A(4,4)$ ,  $B(4)$ . Вычислить  $C=AB$ . Из первой строки  $A$  вычесть  $B$ , из второй строки  $A$  вычесть вектор  $C$ . Печатать  $A$ ,  $B$ ,  $C$ .
29. Заданы массивы  $A(4,4)$ ,  $B(4)$ . Найти максимальный и минимальный элементы  $B$  и поменять их местами. Вычислить  $C=AB$ . Печатать  $A$ ,  $B$ ,  $C$ .
30. Задан массив  $B(4)$ . Вычислить элементы массива  $C(4)$  по формуле:  $c_i = i - \sin(0.4i + 1)$ . Найти и поменять местами максимальные элементы  $B$  и  $C$ .

Вычислить элементы массива  $D(4,4)$  по формуле  $d_{ij} = \sqrt{|b_i + c_j|}$ .

Печатать B, C, D.

#### Задачи к защите лабораторной работы № 4

1. Даны  $A(n, n+1)$ ,  $B(n+1)$ ,  $C(n+1)$ . Получить  $D(n+1, n+2)$  вставкой в  $A$  после строки с номером  $p$  вектора  $B$  и последующей вставкой после столбца с номером  $g$  вектора  $C$ .
2. Дана  $A(n, m)$ . Получить  $D(n, m)$  перестановкой в  $A$  столбцов – первого с последним, второго с предпоследним и т.д.
3. Дана  $A(n, m)$ . Получить  $D(n, m)$  перестановкой в  $A$  строк – первой с последней, второй с предпоследней и т.д.
- 4-9. Дана  $A(n, m)$ . Получить  $D(n, m)$ . Каждый элемент  $d_{ij}$  равен сумме элементов матрицы  $A$  в заштрихованной области



10. Дана  $A(n, m)$ . Получить  $D(n-1, m-1)$  путем выбрасывания из  $A$  строки и столбца, на пересечении которых расположен максимальный элемент  $A$ .
11. Дана действительная матрица  $A(n, m)$ . Найти сумму наибольших значений элементов ее строк.
12. Дано действительное число  $x$ . Получить  $A(n, n)$ :

$$\begin{bmatrix} 1 & x & x^2 & \dots & x^{n-1} \\ x^{n-1} & 1 & x & \dots & x^{n-2} \\ x^{n-2} & x^{n-1} & 1 & \dots & x^{n-3} \\ \dots & \dots & \dots & \dots & \dots \\ x & x^2 & x^3 & \dots & 1 \end{bmatrix}$$

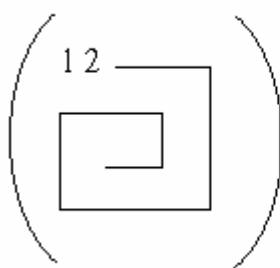
13. Дан вектор  $B(n)$ . Получить  $A(n,n)$ :

$$\begin{bmatrix} b_1 & b_2 & b_3 & \dots & b_n \\ b_2 & b_3 & b_4 & \dots & b_1 \\ \dots & \dots & \dots & \dots & \dots \\ b_n & b_1 & b_2 & \dots & b_{n-1} \end{bmatrix}$$

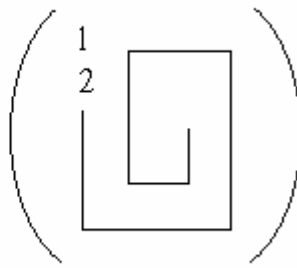
14. Дан вектор  $B(n)$ . Получить  $A(n,n)$ :

$$\begin{bmatrix} b_1 & b_2 & b_3 & \dots & b_n \\ 0 & b_1 & b_2 & \dots & b_{n-1} \\ \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & b_1 \end{bmatrix}$$

15-16. Получить матрицу  $A(7,7)$ , элементами которой являются числа  $1,2,3,4,\dots,49$



15



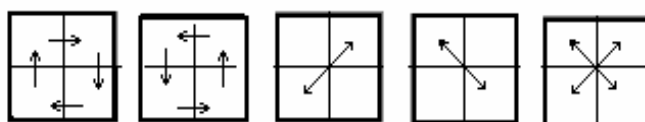
16

17-25. Дана матрица  $A(n,n)$ . Найти наибольший ее элемент в заштрихованной области



17      18      19      20      21      22      23      24      25

26-30. Дана матрица  $A(2n, 2n)$ . Получить  $D(2n, 2n)$ , переставляя блоки  $n \times n$  матрицы  $A$ :



26      27      28      29      30

### Литература

1. Бартенев О.В. Современный FORTRAN. М.: Диалог-МИФИ, 2000.- 447 с.
2. Бартенев О.В. FORTRAN для студентов. М.: Диалог-МИФИ, 1999.- 400 с.
3. Рыжиков Ю.И. Программирование на Фортране PowerStation для инженеров. Практическое руководство. С.-Петербург, Корона принт, 1999.- 159 с.
4. Бартенев О.В. Visual Fortran: новые возможности. М.: Диалог-МИФИ, 1999.- 301 с.
5. Штыков В.В. FORTRAN & WIN32 API. Создание программного интерфейса для Windows средствами современного Фортрана. М.: Диалог-МИФИ, 2001.- 302 с.
6. Мудров А.Е. Численные методы для ПЭВМ на языках Бейсик, Фортран и Паскаль. Томск: Раско, 1992.- 270 с.

## Задача 5. РЕШЕНИЕ НЕЛИНЕЙНЫХ УРАВНЕНИЙ.

Решение нелинейных (трансцендентных или алгебраических) уравнений вида  $f(x)=0$  заключается в отыскании одного или нескольких корней, т.е. таких значений аргумента  $x$ , для которых функция  $f(x)$  обращается в нуль. В общем случае функции  $f(x)$  не имеют аналитических формул для своих корней, поэтому приходится использовать приближенные методы.

Решение нелинейных уравнений обычно состоит из двух этапов:

- 1) Отделение, или локализация корней, т.е. отыскание таких отрезков  $[a,b]$  (одного или нескольких), внутри которых имеется только один корень нелинейного уравнения.
- 2) Уточнение приближенного значения корня до некоторой заданной степени точности.

Отделить корни в некоторых случаях можно графически. Так, если уравнение имеет вид  $\cos(x)-x=0$ , то переписав его в виде  $\cos(x)=x$  и построив графики функций  $y_1=\cos(x)$  и  $y_2=x$ , найдем приближенное значение корня как точку пересечения функций  $y_1$  и  $y_2$ .

Локализовать корень уравнения можно также программно, вычисляя функцию  $f(x)$  для значений  $x$ , изменяющихся с некоторым заданным шагом  $h$ . При этом отыскиваются два таких соседних значения  $x$ , для которых  $f(x)$  имеет противоположные знаки. Алгоритм отделения корня удобно оформить в виде подпрограммы. В качестве параметров подпрограммы можно выбрать следующие:  $F, XL, XR, H, A, B, IER$ .

Входные параметры:

$F$  - имя внешней функции  $f(x)$ ;

$XL, XR$  - соответственно левая и правая границы отрезка оси  $x$ , на котором отделяется корень уравнения;

$H$  - шаг перебора аргумента функции.

Выходные параметры:

$A, B$  - соответственно левая и правая границы отрезка  $[a,b]$ , содержащего первый, считая от точки  $XL$  вправо, корень нелинейного уравнения;

$IER$  - код ошибки.  $IER=0$ , если на участке от  $XL$  до  $XR$  с шагом  $H$  найден отрезок  $[A,B]$ , содержащий корень уравнения;  $IER=1$ , если корень не локализован.

Алгоритм отделения корня нелинейного уравнения может быть следующим:

- 1) Зададим  $IER=1$ , т.е. предполагаем, что на отрезке от  $XL$  до  $XR$  корней нет. В дальнейшем проверим, так ли это.
- 2) Зададим начальное значение аргумента функции  $X=XL$ .

- 3) Вычислим значение функции в точке  $X$ :  $Y=F(X)$ .
- 4) Изменим значение аргумента функции на величину шага :  $X=X+H$ .
- 5) Проверим, лежит ли  $X$  внутри  $[X_L, X_R]$ : если  $X > X_R$ , то возвращаемся в вызывающую программу. В этом случае отрезок  $[A, B]$  не найден,  $IER=1$ .
- 6) Вычислим значение функции в новой точке  $X$ :  $Z=F(X)$ .
- 7) Проверим, изменила ли функция знак при последнем изменении аргумента. Если не изменила, то опять меняем  $X$ : если  $Y*Z > 0$  идти на 4.
- 8) В противном случае определим границы отрезка  $[A, B]$  по формулам  $A=X-H$ ,  $B=X$ , изменим код ошибки  $IER=0$  и вернемся в вызывающую программу.

Для уточнения корня нелинейного уравнения до заданной точности можно воспользоваться методами половинного деления, Ньютона, простых итераций и другими.

МЕТОД ПОЛОВИННОГО ДЕЛЕНИЯ состоит в построении последовательности вложенных отрезков, на концах которых функция принимает значения разных знаков. Каждый последующий отрезок получают делением предыдущего пополам. Процесс построения последовательности отрезков позволяет найти корень уравнения  $f(x)=0$  с любой заданной точностью.

В качестве параметров подпрограммы метода половинного деления можно выбрать следующие:  $F, A, B, EPS, C, N, IER$ .

Входные параметры:

$F$  - имя внешней функции  $f(x)$ ;

$A, B$  - соответственно левая и правая граница отрезка, содержащего один корень уравнения  $f(x)=0$ ;

$EPS$  - точность вычисления корня.

Выходные параметры:

$C$  - корень уравнения (если он найден);

$N$  - количество итераций, которое потребовалось выполнить для вычисления корня с заданной точностью;

$IER$  - код ошибки.  $IER=0$ , если корень найден;  $IER=1$ , если  $A > B$  или на отрезке  $[A, B]$  нет корня, т.е. если  $F(A)*F(B) > 0$ .

Алгоритм метода половинного деления:

- 1) Зададим код ошибки  $IER=1$ .
- 2) Проверим, правильно ли заданы исходные данные: если  $F(A)*F(B) > 0$  или  $A > B$ , то возвращаемся в вызывающую программу.



- 3) Зададим текущие границы отрезка, содержащего корень  $A1=A$ ,  $B1=B$ ; зададим начальное число итераций  $N=0$  и  $IER=0$ ; вычислим значение функции в точке  $A1$ :  $Y=F(A1)$ .
- 4) Вычислим середину отрезка  $[A1,B1]$ :  $C=.5*(A1+B1)$ ; изменим счетчик итераций на единицу :  $N=N+1$ .
- 5) Если длина текущего отрезка  $[A1,B1]$  меньше заданной точности, то возвращаемся в вызывающую программу.
- 6) Проверим, внутри какого из отрезков  $[A1,C]$  или  $[C,B1]$  лежит корень уравнения. Если корень лежит внутри  $[A1,C]$ , то изменяем правую текущую границу отрезка, содержащего корень, в противном случае - левую: если  $F(A1)*F(C)<0$ , то  $B1=C$ , иначе  $A1=C$ ; итерационный процесс повторяется с пункта 4.

Метод половинного деления - наиболее универсальный метод отыскания корней нелинейных уравнений. К его недостаткам относится невысокая скорость сходимости.

МЕТОД НЬЮТОНА (метод касательных) является одним из наиболее эффективных методов нахождения корней нелинейных уравнений. Он состоит в построении итерационной последовательности  $x_{i+1}=x_i-f(x_i)/f'(x_i)$ , сходящейся к корню уравнения  $f(x)=0$ . Геометрическая интерпретация метода следующая: если через точку с координатами  $(x_i, f(x_i))$  провести касательную к функции  $y=f(x)$ , то точка пересечения этой касательной с осью абсцисс принимается за уточненное значение корня  $x_{i+1}$ . Метод Ньютона особенно эффективен, когда известно хорошее приближение корня и в окрестности корня функция имеет большую крутизну.

В качестве параметров подпрограммы метода Ньютона можно рекомендовать следующие:  $F, PF, X0, EPS, NMAX, X, N, IER$ .

Входные параметры:

$F$  - имя внешней функции  $f(x)$ ;

$PF$  - имя внешней функции  $f'(x)$ ;

$X0$  - начальное приближение для корня уравнения;

$EPS$  - точность вычисления корня;

$NMAX$  - максимальное количество итерационных циклов.

Выходные параметры:

$X$  - корень уравнения;

$N$  - количество итераций, выполненных для вычисления корня с заданной точностью;

$IER$  - код ошибки. Если корень найден, то  $IER=0$ ; если корень не найден за  $NMAX$  итерационных циклов, то  $IER=1$ .

Используемый в подпрограмме метода Ньютона алгоритм:

- 1) Зададим: IER=1; начальное значение количества итераций N=0; текущее значение корня X=X0.
- 2) Выполним итерационный цикл: изменим содержимое счетчика итераций на единицу N=N+1; если количество итераций превысило NMAX, осуществим возврат в вызывающую программу; вычислим уточненное значение корня по формуле  $Y=X-F(X)/PF(X)$ ; вычислим модуль разности между уточненным и текущим значениями корня  $E=ABS(Y-X)$ ; в качестве нового текущего зададим уточненное значение корня  $X=Y$ ; проверим, следует ли продолжать уточнение корня: если  $E>EPS$ , то итерационный цикл следует повторить с начала пункта 2.
- 3) Если точность вычисления корня достигнута, то следует изменить код ошибки IER=0 и осуществить возврат в вызывающую программу.

Один из недостатков метода Ньютона состоит в том, что пользуясь им, приходится дифференцировать функцию  $f(x)$ . Если сделать это затруднительно, то можно производную заменить конечной разностью.

МЕТОД ПРОСТЫХ ИТЕРАЦИЙ состоит в замене исходного уравнения  $f(x)=0$  эквивалентным ему уравнением  $x=p(x)$  и в построении последовательности  $x_{i+1}=p(x_i)$ , сходящейся к точному решению. Достаточным условием сходимости метода является  $|p'(x)|<1$ , которое должно выполняться для всех  $x$ . При использовании метода простых итераций основную трудность представляет выбор функции  $p(x)$ , удовлетворяющей условию сходимости. При этом, если уравнение имеет несколько корней, то для вычисления каждого из них необходимо выбирать свой вид функции  $p(x)$ . Например, уравнение  $\ln(1.5*x)-2*x+4=0$  имеет два корня. Переписав его как  $\ln(1.5*x)=2*x-4$ , можно получить  $1.5*x=\exp(2*x-4)$ , тогда  $x=\exp(2*x-4)/1.5$  и  $p_1(x)=\exp(2*x-4)/1.5$ . Используя функцию  $p_1(x)$  вычислим первый, близкий к нулю, корень. Если исходное уравнение записать как  $2*x=\ln(1.5*x)+4$ , то получим функцию  $p_2(x)=(\ln(1.5*x)+4)/2$ , применяя которую вычислим второй корень. При составлении подпрограммы метода простых итераций можно использовать алгоритм метода Ньютона, определяя уточненное значение корня по формуле  $Y=P(X)$ , где P - имя внешней функции  $p(x)$ .

Перед составлением головной программы необходимо провести графический анализ поведения функции  $y=f(x)$ . При этом определяется количество нулей функции. Функция может иметь один, несколько или бесконечное множество нулей. В каждом конкретном случае при решении нелинейного уравнения  $f(x)=0$  ставится своя задача, которая может быть

сформулирована как: а) найти единственный корень; б) найти первый положительный корень; в) найти первые М корней, начиная с точки XL; г) найти все корни на отрезке [ XL, XR ] и т.д.. В зависимости от поставленной задачи готовятся исходные данные и составляется алгоритм решения.

Рассмотрим пример. Пусть решается уравнение  $\text{tg}(x) - \text{ctg}(x) + x = 0$ . Это уравнение имеет бесчисленное множество корней. Сформулируем задачу так: найти все корни уравнения на отрезке [ -1.5, 4 ]. Алгоритм программы может быть следующим:

- 1) Описываются внешние функции, являющиеся фактическими параметрами при обращении к процедурам.
- 2) Задаются исходные данные ( отрезок [ XL, XR ], на котором ищутся корни, точность вычисления корней EPS ). С помощью подпрограммы локализации корней на промежутке от XL до XR определяется первый от точки XL отрезок [ A, B ], на котором функция меняет знак. Подпрограмма локализации корня может и не найти такого отрезка. Поэтому после работы подпрограммы проверяется ее код ошибки: если IER=1, то необходимо выдать на печать сообщение о том, что отрезок [A,B] не найден и остановить выполнение программы.
- 4) В дальнейшем, при поиске второго, третьего и т.д. корней отделение корней необходимо проводить не с исходной точки XL, а начиная с правой границы предыдущего отрезка, на котором функция сменила знак. Поэтому необходимо выполнить операцию  $XL=B$ .
- 5) Обращаясь к подпрограмме метода половинного деления, можно уточнить до заданной точности точку смены знака функцией  $f(x)$ . Рассматриваемая функция имеет не только нули, но и точки разрыва, которые отделяются аналогично нулям. После обращения к подпрограмме метода половинного деления следует проверить, является ли найденная точка С корнем уравнения или нет. Это можно сделать, вычислив значение модуля функции в точке С или анализируя поведение производной на концах отрезка [A,B]. Если С - точка разрыва, то необходимо передать управление на пункт 3 и определить следующий отрезок [ A, B ], на котором функция меняет знак.
- 6) В соответствии с заданием в программе должно быть предусмотрено уточнение корней методами Ньютона и простых итераций. В качестве начального приближения для этих методов можно взять середину отрезка [ A, B ].
- 7) После вычисления и печати значения очередного корня управление передается на п. 3.

**ЗАДАНИЕ.** Используя методы половинного деления и Ньютона, найти корни нелинейных уравнений  $f_1(x)=0$  и  $f_2(x)=0$ . Методом простых итераций найти корни

уравнения  $f_2(x)$ . Функции  $f_1(x)$  и  $f_2(x)$  выбрать из таблицы по номеру варианта.

При выполнении задания требуется:

- 1) Построить графики функций  $y_1=f_1(x)$  и  $y_2=f_2(x)$  для того, чтобы определить количество корней и отрезки, где они расположены. Функции могут иметь один, несколько или бесчисленное множество корней. Необходимо решить вопрос с преподавателем, какие корни подлежат определению.
- 2) Составить все необходимое для расчетов программное обеспечение: подпрограммы-функции для вычисления  $f_1(x), f_1'(x), f_2(x), f_2'(x), p_1(x), p_2(x)$ ; подпрограммы для локализации корней, методов половинного деления, Ньютона, простых итераций; головную программу.
- 3) Провести счет. Результаты оформить в виде таблицы. Предусмотреть печать не только окончательных, но и промежуточных результатов (отрезков  $[A, B]$ , числа итераций, кодов ошибок и т.д.).

#### ВАРИАНТЫ ЗАДАНИЙ

N	$f_1(x)$	$f_2(x)$
1	$x + \sqrt{x} + x^{1/3} - 2.5$	$\lg(1.5773 \cdot x) - 2.3041 \cdot x$
2	$\cos(2/x) - 2 \cdot \sin(1/x) + 1/x$	$\ln(7.6221 \cdot x) - 8.591 \cdot x + 1.5$
3	$\cos(x) - \exp(-x^2/2) + x - 1$	$9.33 \cdot \sin(6.977 \cdot x) - 7.25 \cdot x$
4	$\sin(x^2) + \cos(x^2) - 10 \cdot x$	$0.9737 \cdot \exp(-.5067 \cdot x) - x$
5	$\lg(x/2) - \operatorname{ctg}(x/2) + x$	$\lg(2.2083 \cdot x) - 3.2258 \cdot x$
6	$x - \sin(x) - .25$	$\ln(6.0976 \cdot x) - 6.872 \cdot x + 2.01$
7	$\sqrt{x} - \cos(.387 \cdot x)$	$7.67 \cdot \sin(5.983 \cdot x) - 6.01 \cdot x$
8	$x^2 + 4 \cdot \sin(x)$	$0.9286 \cdot \exp(-.5185 \cdot x) - x$
9	$1.8 \cdot x^2 - \sin(10 \cdot x)$	$\lg(3.7855 \cdot x) - 5.5301 \cdot x$
10	$3 \cdot x - \cos(x) - 1$	$\ln(4.5732 \cdot x) - 3.963 \cdot x + 5.154$
11	$2 \cdot x - \ln(x) - 7$	$6.67 \cdot \sin(5.387 \cdot x) - 5.25 \cdot x$
12	$x^2 \cdot \cos(2 \cdot x) - 1$	$0.5458 \cdot \exp(-.5391 \cdot x) - x$
13	$\sin(x + \pi/3) - .5 \cdot x$	$\lg(9.1483 \cdot x) - 13.364 \cdot x$
14	$\cos(x + \pi/8) - x^3$	$\ln(3.9634 \cdot x) - 4.4868x + 2.01$
15	$2 \cdot \operatorname{arctg}(x) - 3 \cdot x + 2$	$5.67 \cdot \sin(4.794 \cdot x) - 4.55 \cdot x$
16	$3 \cdot x - \cos(x) - 1$	$0.7593 \cdot \exp(-.5683 \cdot x) - x$
17	$(x-3) \cdot \cos(x) - 1$	$\lg(5.9937 \cdot x) - 8.7558 \cdot x$
18	$\sqrt{1-x} - \operatorname{tg}(x)$	$\ln(3.0488 \cdot x) - 3.436 \cdot x + 2.5$
19	$\operatorname{ctg}(1.05 \cdot x) - x^2$	$4.33 \cdot \sin(4.008 \cdot x) - 3.55 \cdot x$

20	$.6*3**x-2.3*x-3$	$0.5909*\exp(-.6286*x)-x$
----	-------------------	---------------------------

## 6. ВЫЧИСЛЕНИЕ ОПРЕДЕЛЕННЫХ ИНТЕГРАЛОВ.

Рассматривается задача вычисления определенного интеграла  $I = \int_a^b f(x)dx$ , которая с геометрической точки зрения состоит в определении площади фигуры, ограниченной ординатами  $a$  и  $b$ , осью абсцисс  $x$  и графиком подынтегральной функции  $y=f(x)$ . Обыкновенный определенный интеграл, у которого известна его первообразная  $F(x)$ , вычисляется по формуле Ньютона-Лейбница  $I = F(b) - F(a)$ . Численное интегрирование применяется, если нахождение  $F(x)$  сложно или невозможно. Оно заключается в интерполяции  $f(x)$  на отрезке  $[a,b]$  подходящим полиномом, для которого определенный интеграл вычисляется по формулам численного интегрирования. Обычно отрезок  $[a,b]$  разбивается на  $n$  промежутков  $[x_i, x_{i+1}]$ , где  $x_i = a + i \cdot h$ ,  $i=0,1,\dots,n-1$ ,  $h=(b-a)/n$  - шаг разбиения. Применяя к каждому промежутку  $[x_i, x_{i+1}]$  соответствующие простые формулы, при суммировании получают составные формулы численного интегрирования.

**МЕТОД ПРЯМОУГОЛЬНИКОВ.** На каждом промежутке  $[x_i, x_{i+1}]$  функция  $y=f(x)$  заменяется интерполяционным многочленом нулевого порядка, построенным по значению функции в средней точке  $x_i + h/2$ . Применяя формулу прямоугольников к каждому промежутку и суммируя, получим

$$I = h \cdot \sum_{i=0}^{n-1} f(x_i + h/2) + R(f)$$

где  $R(f)$  - остаточный член, который не используется в вычислениях, но по которому можно судить о точности применяемой формулы. Для метода прямоугольников  $R(f) = h^2 \cdot (b-a) \cdot f''(t)/24$ ,  $f''(t)$  - значение второй производной  $f(x)$  в точке  $x=t$ , где она максимальна.

**МЕТОД ТРАПЕЦИЙ** заключается в линейной аппроксимации  $f(x)$  на промежутке  $[x_i, x_{i+1}]$ . С учетом суммирования смежных ординат внутри отрезка  $[a,b]$  обобщенная формула метода трапеций принимает вид

$$I = h \cdot \{ [f(a) + f(b)] / 2 + \sum_{i=1}^{n-1} f(x_i) \} - R(f),$$

где  $R(f) = h^2 \cdot (b-a) \cdot f''(t)/12$ .

**МЕТОД СИМПСОНА** (парабол) является частным случаем метода Ньютона-Котеса. Последний основан на интерполяции  $f(x)$  в  $m$  промежутках ( т.е. на участке  $[x_i, x_{i+m}]$  ) полиномом Лагранжа. При этом  $f(x)$  должна задаваться  $(m+1)$  ординатами. Формулы

интегрирования точны, если  $f(x)$  - многочлен  $m$ -ой степени. При  $m=1$  получаем метод трапеций, при  $m=2$  - метод Симпсона. Интеграл от  $x_i$  до  $x_{i+2}$  от полинома Лагранжа, проходящего через точки  $(x_i, f(x_i))$ ,  $(x_{i+1}, f(x_{i+1}))$ ,  $(x_{i+2}, f(x_{i+2}))$ , равен  $[f(x_i) + 4*f(x_{i+1}) + f(x_{i+2})] * h/3$ . При суммировании получим обобщенную формулу Симпсона

$$I = h / 3 * \{f(a) + f(b) + 4 * \sum_{i=1,3,\dots}^{n-1} f(x_i) + 2 * \sum_{i=2,4,\dots}^{n-2} f(x_i)\} - R(f),$$

где  $R(f) = n^5 * h^5 * f'(t)/90$ .

Формулу можно представить в следующем виде

$$I = h / 3 * \{f(a) + f(b) + \sum_{i=1}^{n-1} [3 + (-1)^{i+1}] * f(x_i)\} - R(f)$$

При использовании приведенных формул необходимо помнить, что число разбиений  $n$  в методе Симпсона должно быть четным.

МЕТОД ГАУССА основан на интерполяции  $f(x)$  полиномом Лагранжа, но абсциссы  $x$  выбираются из условия обеспечения минимума погрешности интерполяции. Кадратурная формула Гаусса имеет вид

$$I = \frac{b-a}{2} * \sum_{i=1}^n [A_i * f(x_i)] + R(f), \quad x_i = \frac{b+a}{2} + \frac{b-a}{2} * t_i,$$

$t_i$  - узлы квадратурной формулы Гаусса,  $A_i$  - гауссовы коэффициенты,  $n$  - число узлов квадратурной формулы. Узлы  $t_i$  являются корнями многочленов Лежандра степени  $n$  и расположены симметрично на отрезке  $[-1, 1]$ . Гауссовы коэффициенты  $A_i$  равны для симметрично расположенных узлов. Например, для  $n=8$   $t_1 = -t_8 = -0.96028986$ ,  $t_2 = -t_7 = -0.79666648$ ,  $t_3 = -t_6 = -0.52553242$ ,  $t_4 = -t_5 = -0.18343464$ ,  $A_1 = A_8 = 0.10122854$ ,  $A_2 = A_7 = 0.22238104$ ,  $A_3 = A_6 = 0.31370664$ ,  $A_4 = A_5 = 0.36268378$ . Метод Гаусса обеспечивает наивысший алгебраический порядок точности - формула точна для многочленов степени не выше  $(2n-1)$ .

**ЗАДАНИЕ.** Вычислить определенный интеграл от функции  $f(x)$  на отрезке  $[a, b]$  в соответствии с номером варианта (см.таблицу) методами прямоугольников, трапеций, Симпсона, Гаусса ( $n=8$ ). Определить минимально необходимое число разбиений  $n$  отрезка  $[a, b]$  для вычисления интеграла с заданной точностью методами прямоугольников, трапеций, Симпсона.

Порядок выполнения задания.

- 1) Составить программное обеспечение: подпрограммы вычисления интеграла методами прямоугольников, трапеций, Симпсона, Гаусса; подпрограмму-функцию для вычисления значений  $f(x)$ .
- 2) Составить головную программу для исследования точности вычисления интеграла разными методами.
- 3) Выбрать исходные данные и провести счет.

Рекомендации. В качестве формальных параметров подпрограмм вычисления определенного интеграла по методу прямоугольников, трапеций, Симпсона можно выбрать  $A, B, F, N, Y$ .

Входные параметры:

$A, B$  - нижний и верхний пределы интегрирования;

$F$  - имя внешней функции  $f(x)$ ;

$N$  - число разбиений отрезка  $[A, B]$ .

Выходной параметр:  $Y$  - значение интеграла.

Для метода Гаусса параметр  $N$  отсутствует.

Алгоритм метода Симпсона (выбран как пример) может быть записан следующим образом.

- 1) Вычисляется значение шага  $H=(B-A)/N$ ; задается начальное значение суммы  $Y=0$ ; задается начальное значение переменной  $C=-1$  (используется для вычисления  $(-1)^{i+1}$ ).
- 2) В цикле по  $I$  от 1 до  $N-1$  вычисляются: текущее значение аргумента функции  $X=A+I*H$ ; текущее значение переменной  $C$ :  $C=-C$ ; накапливается сумма  $Y=Y+(3+C)*F(X)$  (множитель при  $F(X)$  равен 4 для нечетных точек и 2 - для четных).
- 3) По завершении цикла определяется приближенное значение интеграла по формуле  $Y=(F(A)+F(B)+Y)*H/3$  и осуществляется возврат в вызывающую программу.

В качестве исходных данных для головной программы рекомендуется выбрать пределы интегрирования  $A$  и  $B$ , начальное число разбиений  $N$ , шаг изменения числа разбиений  $N1$ , точность  $EPS$ . При начальном значении  $N$  вычисляются значения интеграла методами прямоугольников, трапеций, Симпсона, Гаусса. Затем  $N$  изменяется на величину шага  $N=N+N1$  и вычисляются новые значения интеграла всеми методами, за исключением метода Гаусса (результат работы последнего не зависит от  $N$ ). По двум значениям для каждого метода определяется относительная погрешность. Если погрешность для кого-либо метода больше заданного  $EPS$ , то вычисления по этому методу повторяются с увеличенным на  $N1$  числом разбиений. Работа программы завершается, когда всеми методами получен результат с заданной точностью. Промежуточные и конечные результаты выводятся в виде таблицы.

ВАРИАНТЫ ЗАДАНИЯ.



n	f(x)	a	b	n	f(x)	a	b
1	$\cos(x+x^2)$	0	1	2	$\sin(x^2+x/3+1)$	.5	2
3	$\exp(\sin(x))$	0	1	4	$\sin(\exp(x))$	1	2
5	$\exp(\cos(x))$	0	.5	6	$\cos(x)^2$	0	pi/2
7	$\sqrt[3]{\exp(x)}$	-1	1	8	$\sin(x+x^3)$	0	1
9	$\ln(x+1/x)$	.5	1.5	10	$\ln(1+\cos(x))$	-pi/3	pi/3
11	$x*\cos(x^2)$	.5	pi/2	12	$\sin(x)/\sqrt{x}$	.2	1.2
13	$x*\cos(x)^2$	0	pi/2	14	$\text{sh}(x)/x$	.1	1
15	$\text{ch}(x^2)$	0	1	16	$\text{th}(x-1)$	0	3
17	$x*\text{sh}(x)$	1	2	18	$\text{ch}(x^2)*x$	0	1
19	$\text{cth}(x+1)$	.5	2	20	$x^2*\text{th}(x)$	0	2

# ИНФОРМАТИКА И ПРОГРАММИРОВАНИЕ

Методические указания для проведения лабораторных работ  
по курсу “Информатика” для студентов 1-ого курса  
факультета летательных аппаратов

НОВОСИБИРСК  
2008

ББК 73я7  
И 741

Составители: *М.А. Леган*, д-р техн. наук,  
*А.И. Белоусов*, канд. техн. наук,  
*Н.В. Третьякова*, канд. техн. наук,  
*П.Е. Рябчиков*, канд. физ.-мат. наук

Рецензент *С.Д. Саленко* д-р техн. наук

Работа выполнена на кафедре прочности летательных аппаратов

© Новосибирский государственный  
технический университет, 2008

## **ОБЩИЕ ТРЕБОВАНИЕ К ОФОРМЛЕНИЮ ЛАБОРАТОРНЫХ РАБОТ**

1. Лабораторные работы оформляются на листах формата А4 (с одной стороны).
2. Титульный лист (один для всех лабораторных работ) выполняется по стандартной форме (см. приложение на последней странице).
3. Каждая лабораторная работа содержит:
  - номер и наименование работы;
  - постановку задачи и исходные данные;
  - необходимые математические выкладки;
  - алгоритм решения задачи;
  - текст программы;
  - результаты вычислений;
  - анализ результатов.
4. Каждая выполненная и защищенная лабораторная работа подписывается преподавателем.
5. Все скрепленные вместе с титульным листом лабораторные работы сдаются в конце семестра преподавателю.

### **ЛАБОРАТОРНАЯ РАБОТА № 1**

#### **ТАБУЛИРОВАНИЕ ФУНКЦИЙ**

**Цель работы.** При табулировании и построении графиков функций получить начальные навыки программирования.

#### **ЗАДАНИЕ**

Получить таблицы шести функций, заданных аналитическими выражениями, с помощью языка программирования Фортран (FORTRAN). Используя табличный процессор **Microsoft Excel**, построить графики функций. Варианты заданий приведены в табл. 1.

Выбор языка программирования Фортран обусловлен тем, что он предназначен для быстрого счета в различных научно-технических приложениях. Именно на решение научно-технических задач нацелена подготовка специалистов на факультете летательных аппаратов.

## ВВЕДЕНИЕ

Выражение – это формула для получения значения. Выражения образуются из операндов и знаков операций, объединяемых по правилам синтаксиса алгоритмического языка. В качестве операндов выражения используются буквальные и именованные константы, переменные, указатели функций. В зависимости от типа возвращаемого результата выражения подразделяются на арифметические, логические, символьные и производного типа. В первой лабораторной работе рассмотрим правила составления арифметических выражений.

При записи арифметических выражений в Фортране используются следующие арифметические операции (в порядке возрастания приоритета) и функции:

–, + – вычитание и сложение;  
\*, / – умножение и деление;  
\*\* – возведение в степень;

COS(x) – cos(x)	SIN(x) – sin(x)	TAN(x) – tg(x)
COTAN(x) – ctg(x)	ATAN(x) – arctg(x)	SQRT(x) – $\sqrt{x}$
EXP(x) – $e^x$	ALOG(x) – ln(x)	ALOG10(x) – lg(x)
ABS(x) –  x	INT(x) – целая часть x	

$$\sec(x) = \frac{1}{\cos(x)}, \operatorname{cosec}(x) = \frac{1}{\sin(x)}, \operatorname{sh}(x) = \operatorname{SINH}(x), \operatorname{ch}(x) = \operatorname{COSH}(x)$$

Порядок выполнения операций одного и того же приоритета – слева направо (кроме возведения в степень). Для изменения упомянутого выше порядка вычислений используются круглые скобки.

Далее рассмотрим пример выполнения следующего варианта:

1) $y = 0.5 \sin(3x + 2);$	5) $x = 2 \cos(t) \sqrt{ \cos(2t) };$
2) $y = 1 + \sqrt{12 - x^2} - 4x;$	$y = 2 \sin(t) \sqrt{ \cos(2t) };$
3) $y = 1 - 3^x;$	
4) $y = \frac{x}{(1 - (x/2))};$	6) $\rho = \frac{16}{5 - 3 \cos(\varphi)}.$

Следует остановиться на подготовке исходных данных для программы. Вначале обязательно устанавливают область определения функции. Для 1-й функции областью определения является вся числовая ось от  $-\infty$  до  $+\infty$ . Но вследствие периодичности функции синус достаточно построить график в интервале  $0 \leq 3x + 2 \leq 2\pi$  или  $-\frac{2}{3} \leq x \leq \frac{2(\pi-1)}{3}$ . Видно также, что областью изменения функции является интервал  $-0.5 \leq y \leq 0.5$ . Каждая пара значений  $x$  и  $y$  должна быть записана в таблицу результатов счета.

Запишем алгоритм решения первой задачи.

1. Зададим тип и значение неизменяемому параметру  $\pi$ .
2. Опишем тип используемых в программе переменных.
3. Откроем файл для записи значений  $x$  и  $y$ .
4. Определим начальное  $a$  и конечное  $b$  значения переменной  $x$ .
5. Введем количество строк  $n$  для табулирования функции.
6. Вычислим шаг  $h$  цикла для изменения переменной  $x$ .
7. Организуем цикл с управляющей переменной  $i$  целого типа.
8. В цикле вычислим значения переменных  $x$  и  $y$ .
9. Напечатаем на дисплей значения  $x$  и  $y$ .
10. Повторим печать значений  $x$  и  $y$  в файл.
11. Закончим цикл.

В соответствии с алгоритмом составим Фортран-программу:

```
PROGRAM LAB1
REAL, PARAMETER :: PI=3.141593
REAL A, B, H, X, Y
INTEGER I, N
OPEN(1, FILE='LAB1.TXT')
A=-2./3.; B=2.*(PI-1.)/3.
PRINT *, ' N=?'; READ *, N
H=(B-A)/N
DO I=0,N
    X=A+H*I
    Y=0.5*SIN(3.*X+2.)
    PRINT 2, X,Y; WRITE(1, 2) X,Y
END DO
2 FORMAT (' X=',F8.4,3X,'Y=',F8.4)
END PROGRAM LAB1
```

Прокомментируем работу программы:

- вначале после ключевого слова PROGRAM объявляется имя программы, которое обязательно должно начинаться с буквы латинского алфавита или символа \$ (в общем случае ключевое слово PROGRAM с именем программы может отсутствовать);
- первый оператор задает тип и значение именованной константе PI;
- затем идут два оператора, описывающие тип используемых в программе переменных;
- оператор OPEN назначает логический номер 1 файлу «LAB1.TXT». Указанный файл создается в папке проекта в процессе работы программы в момент выполнения оператора OPEN;
- в следующей строке через точку с запятой записаны два оператора, которые вычисляют значения переменных A и B (обратите внимание на десятичные точки в неименованных константах вещественного типа);
- оператор PRINT выводит на дисплей в свободном формате символьную константу, которая заключена в апострофы. Эта последовательность символов является подсказкой для пользователя;
- оператор READ требует ввода с клавиатуры значения целой переменной N. После нажатия клавиши [Enter] переменной N будет присвоено набранное значение;
- следующий оператор вычисляет значение переменной H;
- оператор цикла DO организует повторение вычислений в теле цикла необходимое число раз при изменении управляющей переменной I от 0 до N с шагом 1 (шаг равен 1 по умолчанию, так как его значение не указано);
- операторы в следующих двух строках обеспечивают вычисление арифметических выражений, записанных справа от знака равенства. Результаты присваиваются переменным X и Y;
- оператор PRINT работает под управлением оператора FORMAT с меткой 2. Он выводит на экран сначала символьную константу X=, которая в программе заключена в апострофы, затем числовое значение переменной X по дескриптору F8.4. Далее через три позиции выводятся символьная константа Y=, и значение Y также по дескриптору F8.4, где 8 – общее число позиций для выводимого числа, а 4 – количество позиций для цифр после десятичной точки. В общее число позиций включаются знак отрицательного числа «-» и десятичная точка (ноль перед десятичной точкой, как правило, не печатается);

- оператор WRITE повторяет печать значений X и Y в файл «LAB1.TXT» с логическим номером 1 под управлением оператора FORMAT с меткой 2;
- оператор END DO ограничивает тело цикла;
- последний оператор END PROGRAM LAB1 ограничивает программный модуль, заканчивая его (LAB1 или PROGRAM LAB1 могут быть опущены). Если имя программы не объявлено, то оно не может присутствовать и в операторе END.

#### КРАТКАЯ ИНСТРУКЦИЯ ПО РАБОТЕ В СРЕДЕ ПРОГРАММИРОВАНИЯ

Для запуска составленной программы на счет необходимо прежде всего создать проект. Для этого предлагаем использовать среду **Developer Studio** (мастерскую разработчика), в которой программирование осуществляется на языке **Compaq Visual Fortran (CVF)** версии 6.1. Для входа в среду **Developer Studio** необходимо после нажатия кнопки «Пуск» выбрать из меню “Все программы” пункт **Compaq Visual Fortran 6** и далее подпункт **Developer Studio**, либо другим способом – дважды щелкнуть по иконке **Developer Studio** на рабочем столе.

Далее для создания проекта необходимо выполнить следующую цепочку действий: **File – New – выбрать закладку Projects – Fortran Standard Graphics or QuickWin Application – ввести имя проекта – задать расположение проекта на диске**. В качестве места расположения проекта студентам рекомендуется использовать папку с именем группы, специально созданную в папке FLA на жестком диске C: компьютера. После нажатия кнопки **OK** в появившемся окне нужно выбрать кнопку **QuickWin (multiple windows)** и нажать **Finish** и еще раз **OK**. В результате этих действий создается папка с именем проекта. В этой папке будут размещены файлы с расширениями **.dsp**, **.dsw** и **.opt**, фиксирующие структуру проекта. Сам же проект отобразится в окне с закладкой **File View**. Для сохранения конфигурации проекта можно использовать меню **File – Save All**. Желательно запомнить имя и расположение проекта для того, чтобы при необходимости можно было найти и открыть созданный проект.

Заметим, что уже существующий проект открывается по цепочке **File – Open Workspace – выбор файла с расширением .dsw – Open**, а закрывается командами **File – Close Workspace**.

Создадим теперь файл, в который будет занесен текст программы. Для этого необходимо выполнить действия: **File – New – выбрать за-**



кладку **Files** – выбрать **Fortran Free Format Source File** – активизировать опцию (поставить «галочку») **Add to project** – задать имя файла – **OK**. Созданный файл должен иметь расширение **.f90**. Далее наберем в правом окне текст программы. В процессе набора ключевые слова Фортрана и имена встроенных функций отображаются синим цветом, а комментарии – зеленым. Начальный отступ предыдущей строки наследуется, что облегчает набор структурированных программ в свободном формате. При наборе больших программ желательно периодически сохранять файл, записывая его на диск.

Если же файл с программой уже существует, то для добавления его в проект нужно: в окне **FileView** – выбрать папку **Source Files** – нажать на правую кнопку мышки – **Add Files to Folder** – выбрать тип файлов – выбрать необходимые файлы – **OK**. Чтобы отобразить на экране содержимое файла, достаточно с помощью левой клавиши мышки дважды щелкнуть по нему в окне **File View** (в папке **Source Files**). Можно сначала открыть файл в правом окне **Developer Studio**, а затем включить его в проект. Для этого при нахождении курсора мышки в правом окне нужно нажать правую клавишу мышки, в появившемся контекстном меню выбрать пункт **Insert File into Project** и щелкнуть левой клавишей мышки на имени проекта.

Для удаления файла из открытого проекта достаточно выбрать этот файл в окне **FileView** и нажать клавишу [Del].

Для перевода программы на язык машинного уровня выполним компиляцию проекта с помощью меню **Build – Compile** и исправим обнаруженные ошибки, сообщения о которых появятся в нижнем окне экрана. При отсутствии ошибок программа переводится с языка Фортран на язык машинного (объектного) уровня и создается объектный модуль с расширением **.obj**.

Далее, используя меню **Build – Build**, построим исполняемый файл с расширением **.exe**, в котором объединяются объектные коды главной программы и вызываемых ею вспомогательных процедур, в том числе встроенных.

Исполняемый файл запустим на счет с помощью меню **Build – Execute**. Результат выводится в специальное рабочее окно. Отметим, что кириллические тексты на экране воспроизводятся в искаженном виде. Фрагменты содержимого окна результатов можно переносить в другие файлы через буфер обмена. Прекратить счет можно нажатием комбинации клавиш [Ctrl + C]. Если программа закончила работу, то можно

выйти из окна сразу и вернуться к программе или поработать с результатами средствами операционной системы **Windows** и затем закрыть окно.

Компиляцию, построение исполняемого модуля и запуск программы можно также выполнить с помощью кнопок панели инструментов или клавишных комбинаций, имеющихся в меню **Build**.

### ПОЛУЧЕНИЕ ТАБЛИЦ ОСТАЛЬНЫХ ЗАДАНЫХ ФУНКЦИЙ

Во 2-й задаче область определения функции устанавливается исходя из того, что подкоренное выражение должно быть неотрицательно. Следовательно,  $12 - x^2 - 4x \geq 0$ . Решая квадратное уравнение, получаем область определения:  $-6 \leq x \leq 2$ . При исполнении программы все значения аргумента следует брать только из этой области, так как вне ее функция не определена. Следовательно, в программе для 2-й задачи требуется изменить значения переменных A и B в третьей строке, набрав вместо старых операторов новые

$$A = -6.; B = 2.$$

И в соответствии с заданием изменяется оператор вычисления переменной y :

$$Y=1.+SQRT(12.-X**2-4.*X)$$

Указанные изменения в программе лучше оформить в отдельном файле, который можно получить из прежнего, выполнив **File – Save As** и набрав имя нового файла. Затем следует выполнить удаление из проекта старого файла, включение в проект нового файла, компиляцию, построение загрузочного модуля и запуск программы.

Аналогично проводятся вычисления для 3-й и 4-й функций, для которых также лучше оформить отдельные файлы.

В 5-й задаче функция задается параметрически, поэтому здесь сначала в цикле вычисляется переменная T, соответствующая параметру функции, а затем переменные X и Y. В результате операторы присваивания принимают вид:

$$\begin{aligned} T &= A + H * I \\ X &= 2. * \cos(T) * \sqrt{\text{ABS}(\cos(2. * T))} \\ Y &= 2. * \sin(T) * \sqrt{\text{ABS}(\cos(2. * T))} \end{aligned}$$

В 6-й задаче функция описывается полярными координатами с радиусом-вектором  $\rho$  из начала декартовых координат и углом  $\varphi$ , который отсчитывается от оси абсцисс против часовой стрелки. Целесообразно перейти от полярных координат к декартовым координатам по формулам:

$$x = \rho \cos(\varphi); \quad y = \rho \sin(\varphi).$$

С использованием переменных  $r = \rho$ ,  $f = \varphi$  для вычисления значений  $x$  и  $y$  новую программу лучше сделать из предыдущей, в которой следует заменить переменную  $T$  на переменную  $F$ , а также изменить два оператора присваивания для переменных  $X$ ,  $Y$  и перед ними добавить еще один, соответствующий заданной функции:

```
R=16./(5.-3.*COS(F))
X=R*COS(F); Y=R*SIN(F)
```

В рассмотренном примере угол  $\varphi$  необходимо задавать в интервале от 0 до  $2\pi$ . В общем случае область определения функции находится из условия  $\rho \geq 0$ .

### ВЫЧИСЛЕНИЕ ФУНКЦИЙ И ПОСТРОЕНИЕ ГРАФИКОВ В ТАБЛИЧНОМ ПРОЦЕССОРЕ Microsoft Excel

Перед началом работы необходимо осуществить запуск **Excel**, щелкнув мышью на кнопке <Пуск> и выполнив команду **Все программы / Microsoft Office / Microsoft Office Excel**. Пустая электронная таблица, открываемая **Excel** сразу после его запуска, по умолчанию имеет имя **Book1 (Книга1)**.

#### ВЫЧИСЛЕНИЕ ФУНКЦИЙ

Ввод данных осуществляется в активную ячейку. Активной называется ячейка, в которой в данный момент времени расположен табличный (или окошечный) курсор. Курсор перемещается с помощью клавиатуры или с помощью мыши. Введите данные в пустую электронную таблицу в следующем порядке.

- Сделайте активной ячейку A1 и введите текст «Функция 1». Обратите внимание, что вводимая информация дублируется в строке формул. Для окончания ввода информации в ячейку необходимо или нажать клавишу **Enter**, или щелкнуть по галочке слева от строки редактирования, или перевести курсор в другую ячейку. В ячейки C1, C2, C3, C4 ввести соответственно «N», «A», «B», «H». В смежные справа ячейки D1, D2, D3, D4 задать N, начальное, конечное значения переменной  $x$  и шаг. В ячейку D4, где вычисляется шаг, следует записать  $=(D3-D2)/D1$ .
- Сделайте активной ячейку A2 и введите первое значение  $x$  ( $=D2$ ), затем в активную ячейку A3 – второе значение  $x$  ( $=A2+\$D\$4$ ).

- Переведите курсор мышки в ячейку B2 и введите в нее первое значение  $y$  рассматриваемой функции (например  $=\sin(A2)$ ), затем в ячейку B3 введите второе значение  $y$  ( $=\sin(A3)$ ).

- Дальнейшие вычисления можно существенно упростить, поскольку переменная  $x$  изменяется с постоянным шагом. Для этого нужно выделить (маркировать) две ячейки A3 и B3, содержащие значения  $x$  и  $y$ . Выделение осуществляется буксировкой мышки (при нажатой левой клавише) по выбранным ячейкам. Затем нужно отпустить левую клавишу и поместить курсор мышки в правый нижний угол выделенной области так, чтобы появился крестик черного цвета. Далее следует снова нажать левую клавишу мышки и продолжить буксировку вниз до появления крайнего значения  $x$  из заданного диапазона, после чего можно отпустить левую клавишу мышки.

### ПОСТРОЕНИЕ ГРАФИКОВ

Во-первых, необходимо выделить область таблицы, содержащую данные, по которым нам необходимо построить график. В нашем случае это ячейки, начиная с A2 до B12. Выделение группы ячеек осуществляется буксировкой мышки. Затем нужно использовать команду построения диаграммы «Вставка / Диаграмма» или соответствующую пиктограмму. Далее:

- выбираем тип диаграммы «Точечная» и её вид;
- проверяем диапазон адресов маркированных областей данных;
- вводим название диаграммы и осей;
- выбираем режим «построение диаграммы на имеющемся листе».

При необходимости осуществляется корректировка диаграммы. Для этого нужно активизировать диаграмму щелчком левой кнопки мышки, щелкнуть правой кнопкой мышки в области диаграммы на свободном от её элементов месте и в открывшемся контекстном меню выбрать нужную команду или элемент.

### СОХРАНЕНИЕ ДОКУМЕНТА НА ДИСКЕ

Выполните команду «Файл / Сохранить как». В открывшемся окне в поле ввода «Имя файла» введите название файла «Lab1.xls». Обратите внимание, в какой папке будет сохранен документ (при необходимости откройте другую папку). Наконец, нажмите командную кнопку «Сохранить». Аналогично внесите данные и постройте графики остальных пяти функций на отдельных листах книги **Excel**. Добавить листы можно с помощью меню **Вставка – Лист**.

Таблица 1

Варианты заданий к лабораторной работе № 1

№ п/п	1	2	3	4	5	6
1	$y = 2 \sin\left(x/3\right)$	$y =  x + 1  -  x - 1 $	$y = 2 + \sqrt{4 - x^2 - 6x}$	$y = -2 + \frac{2}{3}\sqrt{x^2 - 9}$	$x = \arcsin(t)$ $y = \operatorname{arctg}(t)$	$\rho = \varphi$
2	$y = 2 \sin(3x + \pi)$	$y = 2 x  - x + 1$	$y = -1 + \sqrt{2 - x^2 - 5x}$	$y = \frac{x + 1}{x + 2}$	$x = \cos(t)$ $y = \sin^2(t)$	$\rho = \frac{\pi}{\varphi}$
3	$y = \operatorname{tg}(2x + 1)$	$y = x -  x $	$y = 2 - \sqrt{4 - x^2 - 6x}$	$y = -3\sqrt{(x + 1)^2 + 1}$	$x = \cos(2t)$ $y = \cos^2(t)$	$\rho = e^{\pi/\varphi}$
4	$y = 2 \cos(3x + 1)$	$y = \ln(\sqrt{x - 4})$	$y = \frac{1}{2} - \sqrt{x^2 - 2y - \frac{1}{4}}$	$y = 1 + \frac{7}{3}\sqrt{3 + x^2 + 4x}$	$x = 1$ $y = \sin(t)$	$\rho = 2\varphi$
5	$y = -\sin\left(\frac{x}{2} - 1\right)$	$y = \sqrt{x^3}$	$y = -1 - \sqrt{2 - x^2 - 5x}$	$y = 3 - \frac{4}{3}\sqrt{x^2 + 9}$	$x = \sqrt{\cos(t)}$ $y = \sqrt{\sin(t)}$	$\rho = \sec^2\left(\frac{\varphi}{2}\right)$
6	$y = 2 \operatorname{tg}(2x) $	$y = \ln(x^2)$	$y = 1 + \frac{3}{4}\sqrt{5 - x^2 - 2x}$	$y = 1 - \frac{4}{3}\sqrt{1 + x^2 + 4x}$	$x = \sin(t)$ $y = \ln(\sin(t))$	$\rho^2 \cos(2\varphi) = 1$
7	$y = \frac{1}{2} \sin 2x $	$y = 2 \ln(x)$	$y = 2 - \frac{5}{2}\sqrt{8 - x^2 + x}$	$y = 1 + \frac{2}{3}\sqrt{x^2 - x + 5}$	$x = \arcsin(t)$ $y = t^2$	$\rho = \frac{2}{1 - \cos(\varphi)}$

Продолжение табл. 1

№ п/п	1	2	3	4	5	6
8	$y = \operatorname{tg} \left  \frac{x}{2} \right $	$y =  x  - 1$	$y = 1 - \frac{2}{3} \sqrt{5 - x^2} - x$	$y = 2 + \frac{3}{4} \sqrt{x^2 - x - 6}$	$x = 2^t$ $y = 2^{2t}$	$\rho = \operatorname{tg} \left( \frac{\varphi}{2} \right)$
9	$y = 2 \sin \left( \frac{\pi}{3} - x \right)$	$y = x +  x $	$y = 1 + \frac{1}{7} \sqrt{8 - x^2} - x$	$y = 2 + \frac{2}{5} \sqrt{x^2 + 25}$	$x = 3^t$ $y = 9^t - 3^t$	$\rho = \operatorname{cosec}^2 \left( \frac{\varphi}{2} \right)$
10	$y =  \cos(3x) $	$y = 2 x - 3  + 5x$	$y = 3 + \frac{3}{5} \sqrt{9 - x^2} + x$	$y = 2 - \frac{3}{4} \sqrt{x^2 - x - 3}$	$x = (t + 1/t)/2$ $y = (t - 1/t)/2$	$\rho = \varphi^2$
11	$y = \operatorname{tg} \left( \frac{x}{2} - 1 \right)$	$y = -\frac{3}{2}x + 1$	$y = 1 + \frac{1}{2} \sqrt{5 - 3x^2} - x$	$y = 7 - \frac{3}{2} \sqrt{x^2 - 6x + 8}$	$x = 5^{t+1}$ $y = 5^{2t+3}$	$\rho = \sin \left( \frac{\varphi}{2} \right)$
12	$y = \frac{3}{2} \cos \left( \frac{x}{2} + 1 \right)$	$y = 2 x  + 3$	$y = -1 + \frac{1}{3} \sqrt{x^2 + 3x}$	$y = 1 + \frac{3}{4} \sqrt{x^2 + 4x + 9}$	$x = 2^{\sin(t)}$ $y = \sin(t)$	$\rho^2 \cos(2\varphi) = 9$
13	$y = \frac{2}{3} \sin(3x + 1)$	$y = \frac{1}{7}(x + 1) - 2$	$y = 2 + \frac{2}{3} \sqrt{4 - x^2} + x$	$y = 9 - 2\sqrt{x^2 + 4x + 8}$	$x = \sqrt{t}$ $y = 1/t$	$\rho = \frac{1}{2 + \cos(\varphi)}$
14	$y =  \operatorname{tg}  2x  $	$y = \sqrt{x^2 - 4x + 4}$	$y = 3 - \frac{3}{5} \sqrt{9 + 5x^2} + x$	$y = 1 - \frac{3}{4} \sqrt{x^2 + 4x + 1}$	$x = \cos(t)$ $y = 1/\sin(t)$	$\rho = \frac{2}{\cos(\varphi)}$

Продолжение табл. 1

№ п/п	1	2	3	4	5	6
15	$y = \operatorname{tg}(1-x)$	$y = \sqrt{x^2 - 6x + 9}$	$y = 1 - \frac{1}{2}\sqrt{6 - x^2 - 2x}$	$y = 5 - \frac{4}{3}\sqrt{x^2 + 4x - 2}$	$x = \sec(t)$ $y = \cos(t)$	$\rho = \frac{2}{1 + \cos(\varphi)}$
16	$y = -\frac{2}{7}\sin(x+1)$	$y = \frac{1}{5}(x-1) + 1$	$y = 1 - \frac{1}{3}\sqrt{-5x^2 + 3x}$	$y = 2 + \frac{4}{3}\sqrt{x^2 + 2x - 1}$	$x = \cos(t)$ $y = \cos(2t)$	$\rho = \frac{8}{3 - \cos(\varphi)}$
17	$y = -3\sin(5x)$	$y = \frac{1}{5}x - 1$	$y = 2 - \frac{2}{3}\sqrt{9 - x^2 + x}$	$y = \frac{1}{x} - \frac{1}{2}$	$x = \operatorname{tg}(t)$ $y = \frac{2}{1 + \cos(t)}$	$\rho = 1 + \cos(\varphi)$
18	$y = \left  \operatorname{ctg}(3x) \right $	$y = \sqrt{(x-1)^2 + 3}$	$y = 1 + \frac{7}{3}\sqrt{x^2 - 4x + 5}$	$y = 2 - \frac{4}{3}\sqrt{x^2 + 2x - 8}$	$x = \sin\left(\frac{\pi}{2} + t\right)$ $y = \sin(\pi + t)$	$\rho^2 = 36\cos(\varphi)$
19	$y = 1 - \cos(3x)$	$y = 1 + \sqrt{x^2 - x + 1}$	$y = 2 - \frac{3}{7}\sqrt{x^2 + 2x + 4}$	$y = \frac{2x}{3 - x}$	$x = \cos(t)$ $y = \pi \sin(t)$	$\rho = 1 - \cos(\varphi)$
20	$y = \frac{1}{3} \left  \cos(2x) \right $	$y = 1 + \sqrt{x^2 - x + 4}$	$y = 1 + \frac{4}{3}\sqrt{-6x - x^2}$	$y = 1 + \frac{3}{2}\sqrt{x^2 + 4x + 8}$	$x = 1 + \operatorname{tg}(t)$ $y = \operatorname{ctg}^2(t)$	$\rho = \cos(2\varphi)$
21	$y = \operatorname{ctg}  3x - 3 $	$y =  x  -  x + 2 $	$y = 1 - \frac{7}{3}\sqrt{x^2 - 4x + 5}$	$y = \frac{-x + 4}{x - 1}$	$x = \operatorname{cosec}(t)$ $y = \operatorname{tg}^2(t)$	$\rho = \sin^3 \varphi$

Продолжение табл. 1

№ п/п	1	2	3	4	5	6
22	$y =  \operatorname{tg}(x)  + 2$	$y = 3 - \sqrt{x^2 - x + 1}$	$y = 7 + \frac{2}{5}\sqrt{6 - 6x - x^2}$	$y = 1 - \frac{3}{2}\sqrt{x^2 + 4x + 8}$	$x = \operatorname{cosec}(t)$ $y = \sin(t)$	$\rho = \sin^2(\varphi)$
23	$y =  \sin(x) - 1 $	$y = \frac{3}{2}x + \frac{1}{2}$	$y = 2 + \frac{3}{7}\sqrt{x^2 + 2x + 8}$	$y = \frac{x - 1}{2x - 5}$	$x = \sec(t)$ $y = \ln(\cos(t))$	$\rho = 4 \sin(\varphi)$
24	$y = \frac{1}{3} \sin \left  \frac{x}{2} \right $	$y = \ln  x $	$y = 2 - \sqrt{-5 - 6x - x^2}$	$y = 2 + \frac{2}{3}\sqrt{x^2 - x + 9}$	$x = \cos(t)$ $y = \lg(\sec(t))$	$\rho = \cos(\varphi) + \varphi$
25	$y =  \operatorname{tg}(3x + 2) $	$y =  \ln(x - 1) $	$y = 5 + \frac{1}{4}\sqrt{8 + 2x - x^2}$	$y = \frac{1 - x}{1 + x}$	$x = t - \sin(t)$ $y = 1 - \cos(t)$	$\rho = \frac{1}{2} + \sin(\varphi)$
26	$y = \operatorname{ctg} \left( x/2 \right) - 1$	$y = e^{ x  - 1}$	$y = 5 + \sqrt{40 - 6x - x^2}$	$y = 2 - \frac{2}{3}\sqrt{x^2 - x + 8}$	$x = \cos^3(t)$ $y = \sin^3(t)$	$\rho = \sin \left( \varphi - \frac{\pi}{4} \right)$
27	$y = \operatorname{ctg}(2x - 1)$	$y = e^{-x^2}$	$y = 3 + \sqrt{5 - x^2 - 4x}$	$y = \frac{3x - 5}{x - 1}$	$x = 5 \cos^2(t)$ $y = 3 \sin^2(t)$	$\rho = 1 + 2 \cos(\varphi)$
28	$y = \left  \sin \left( \frac{x}{2} - 1 \right) \right $	$y = \ln  x + 1 $	$y = -5 - \sqrt{7 - x^2 - 4x}$	$y = 1 + \frac{4}{3}\sqrt{4 + x^2 + 4x}$	$x = 2 \sec^2(t)$ $y = \sin(2t)$	$\rho = 1 - \sin(\varphi)$



Окончание табл. 1

№ п/п	1	2	3	4	5	6
29	$y = \frac{1}{2} \sin  3x + 2 $	$y = 1 - 3^x$	$y = 1 + \sqrt{12 - x^2} - 4x$	$y = \frac{x}{1 - x/2}$	$x = \cos(t)$ $y = \sin(t)$	$\rho = \frac{8}{3 - \cos(\varphi)}$
30	$y = 2 + \operatorname{tg}\left(\frac{x}{3} + 1\right)$	$y = \left(\frac{1}{3}\right)^{ x-1 }$	$y = -2 + \sqrt{-x^2 - 10x}$	$y = 1 - \frac{4}{3} \sqrt{4 + 4x + x^2}$	$x = t^2 / (1 + t^2)$ $y = t / (1 + t^2)$	$\rho = \cos^2\left(\frac{\varphi}{2}\right)$

## ЛАБОРАТОРНАЯ РАБОТА № 2

### ПРОГРАММИРОВАНИЕ ВЕТВЯЩИХСЯ ВЫЧИСЛИТЕЛЬНЫХ ПРОЦЕССОВ

**Цель работы.** Получить навыки программирования ветвящихся вычислительных процессов. Изучить условные операторы, переменные логического типа, логические отношения и операции.

#### ПОСТАНОВКА ЗАДАЧИ

Пусть на плоскости  $x, y$  задана область  $D$ . Требуется определить, «попадает» ли точка с координатами  $(x, y)$  в область  $D$  или нет. Если попадает, то необходимо вычислить функцию  $z = f_1(x, y)$  и переменной  $M$ , значение которой является признаком «попадания» или «непопадания», присвоить значение  $M = 1$ . В противном случае вычислить  $z = f_2(x, y)$  и положить  $M = 2$ .

В каждом варианте задания область  $D$  можно разбить на две подобласти:  $D_1, D_2$ . Подобласть  $D_1$  ограничена прямыми линиями, уравнения которых нужно составить. При этом удобно использовать уравнение прямой в отрезках на осях:  $\frac{x}{a} + \frac{y}{b} = 1$ . Подобласть  $D_2$  ограничена кривой второго порядка, уравнение которой указано в задании. Варианты заданий приведены в таблице 2.

#### ПОЯСНЕНИЕ ЗАДАЧИ

1. Для заданного варианта из таблицы нарисовать область  $D$  и выписать функции  $f_1(x, y)$  и  $f_2(x, y)$ . Записать уравнения, определяющие границы области, и подготовить исходные данные (координаты точек) так, чтобы часть точек находилась внутри области  $D$ , а часть – вне ее (рис. 1). При этом для каждой четверти координатной плоскости рекомендуется выбрать по две точки.
2. Создать на компьютере проект для решения поставленной задачи, составить и отладить программу.
3. Для выбранных точек получить и записать результаты счета.

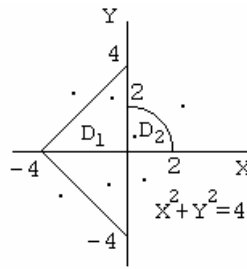


Рис. 1. Область на плоскости  $x, y$

Рассмотрим один из вариантов решения задачи. Пусть область  $D$  имеет вид, показанный на рис. 1, а функции  $f_1$  и  $f_2$  заданы как

$$f_1(x, y) = \ln(|x + y| + 1);$$

$$f_2(x, y) = e^{\frac{1}{|x| + |y|}}.$$

Прямой  $x = 0$  разобьем область  $D$  на две подобласти:  $D_1$  и  $D_2$ . Уравнения прямых, ограничивающих подобласть  $D_1$ , будут: во второй четверти  $y = x + 4$ ; в третьей  $y = -x - 4$ .

Уравнение окружности с центром в начале координат, ограничивающей подобласть  $D_2$  при  $x > 0$  и  $y > 0$ , имеет вид  $x^2 + y^2 = 4$ .

Теперь можно записать, что некоторая точка  $(x, y)$  «попадает» в область  $D$ , если она «попадает» в область  $D_1$  или  $D_2$ , т.е. если  $x \leq 0$ , и  $y \leq x + 4$ , и  $y \geq -x - 4$  или  $x > 0$  и  $y \geq 0$ , и  $x^2 + y^2 \leq 4$ .

Точка  $(x, y)$  «не попадает» в область  $D$ , если

$$x \leq 0 \text{ и } y > x + 4, \text{ или } x \leq 0 \text{ и } y < -x - 4, \text{ или}$$

$$x > 0 \text{ и } y < 0, \text{ или } x > 0 \text{ и } x^2 + y^2 > 4.$$

При выполнении задания можно проводить анализ как «попадания», так и «непопадания» точки в область  $D$ .

При составлении программы на Фортране для анализа «попадания» или «непопадания» точки в область  $D$  можно составить только одно логическое выражение с использованием следующих логических отношений и операций:

отношения:	операции:
(в свободном и фиксированном форматах)	(в порядке убывания приоритета)
< – .LT. – меньше,	.NOT. – отрицание,
<= – .LE. – меньше или равно,	.AND. – конъюнкция («и»),
> – .GT. – больше,	.OR. – дизъюнкция («или»),
>= – .GE. – больше или равно,	.EQV. – эквивалентность,
== – .EQ. – равно,	.NEQV. – неэквивалентность.
/= – .NE. – не равно.	

Результатом вычисления отношений является одно из двух логических значений .TRUE. – истина или .FALSE. – ложь. С помощью логических операций можно создавать сложные логические выражения, которые также могут принимать только два значения .TRUE. или .FALSE.

Например, точка  $(x, y)$  «попадает» в область  $D$  (рис. 1), если логическое выражение

$X \leq 0 \text{..AND.} Y \leq X + 4 \text{..AND.} Y \geq -X - 4 \text{..OR.} X > 0 \text{..AND.} Y \geq 0 \text{..AND.} X^2 + Y^2 \leq 4$ .

принимает значение .TRUE. – истина.

Ветвящиеся вычислительные процессы в Фортране можно запрограммировать с помощью следующих условных операторов IF.

**Логический оператор IF** имеет вид

IF(L) S

Здесь L – логическое выражение, S – исполняемый оператор.

Если логическое выражение L истинно, то выполняется оператор S; если оно ложно, то выполнится следующий за IF оператор, а S не выполняется.

**Блочный оператор IF** имеет вид

```
IF(L) THEN
    < блок операторов 1 >
ELSE
    < блок операторов 2 >
END IF
```

Правило выполнения: если логическое выражение L истинно, то выполняется < блок операторов 1 >, а если оно ложно, то выполняется < блок операторов 2 >. При отсутствии альтернативы < блок операторов 2 > вместе с ELSE опускается.

При необходимости дополнительных условий используется общий вид блочного оператора IF:

```
IF(L) THEN
    < блок операторов 1 >
ELSE IF(L+) THEN
    < дополнительный блок операторов >
ELSE
    < блок операторов 2 >
END IF
```

Здесь L и L+ – основное и дополнительное логические выражения.

Правило выполнения: если логическое выражение L истинно, то выполняется < блок операторов 1 >; если выражение L ложно, а L+ истинно, то выполняется < дополнительный блок операторов >; если выражение L ложно и L+ ложно, то выполняется < блок операторов 2 >. Дополнительных условий, соответствующих им логических выражений и блоков операторов может быть несколько.

Приведем пример Фортран-программы, в которой для анализа «попадания» или «непопадания» точки используется логическая переменная C. Этой переменной присваивается значение логического выражения. Если C=.TRUE., то в блочном операторе IF (C) THEN выполняется первый блок операторов, а если C=.FALSE., то второй блок (стоящий после ELSE).

```
PROGRAM LAB2
LOGICAL C
REAL X, Y, Z
INTEGER I, M
OPEN(1, FILE='LAB2.TXT')
DO I=1,8
    PRINT *, ' X=?, Y=? '; READ *, X,Y
    C=X<=0..AND.Y<=X+4..AND.Y>=-X-4..OR.&
    &X>0..AND.Y>=0..AND.X**2+Y**2<=4.
    IF(C) THEN
        PRINT *, ' YES '
        Z=F1(X,Y); M=1
```

```

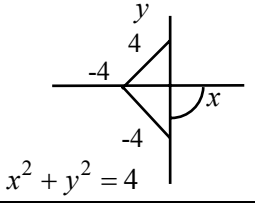
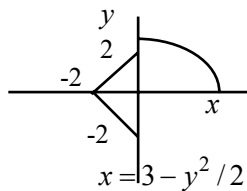
ELSE
    PRINT *, 'NO '
    Z=F2(X,Y); M=2
END IF
PRINT 2, X,Y,Z,C,M; WRITE(1, 2) X,Y,Z,C,M
END DO
2   FORMAT (' X=,F7.3,3X,'Y=,F7.3,3X,'Z=,E11.4,3X,'C=',L2,&
&3X,'M=',I2/)
END PROGRAM LAB2
FUNCTION F1(X,Y)
REAL F1,X,Y
F1=ALOG(ABS(X+Y)+1.)
END FUNCTION F1
FUNCTION F2(X,Y)
REAL F2,X,Y
F2=EXP(1./(ABS(X)+ABS(Y)))
END FUNCTION F2

```

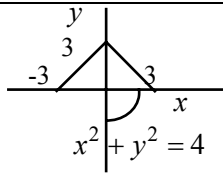
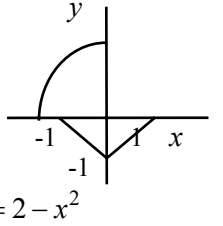
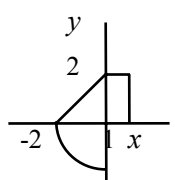
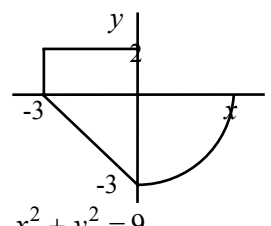
В конце программы описаны внутренние процедуры-функции F1 и F2 . Эта программа для своего завершения требует ввести координаты 8 точек.

Т а б л и ц а 2

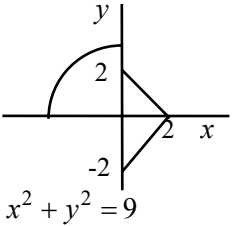
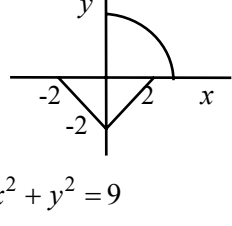
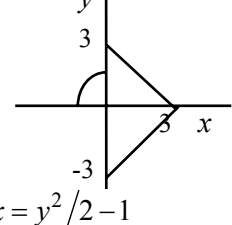
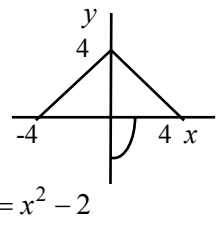
**Варианты заданий к лабораторной работе № 2**

№ п/п	$f_1(x, y)$	$f_2(x, y)$	Область
1	$\operatorname{tg}(x) + \ln(y^2)$	$e^{\sin^2(x) + \cos^2(y)}$	 $x^2 + y^2 = 4$
2	$\frac{\sin(x)}{1 - \cos(y)}$	$e^{\operatorname{arctg}(x^2 + y^2)}$	 $x = 3 - y^2 / 2$

Продолжение табл. 2

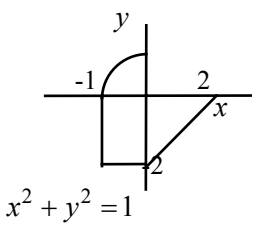
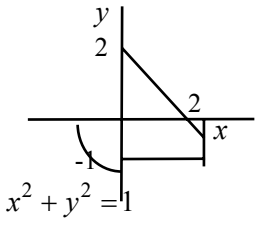
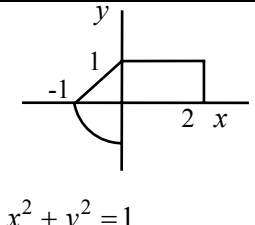
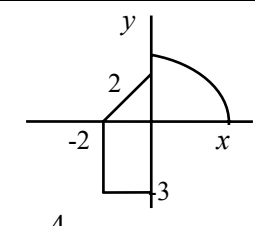
№ п/п	$f_1(x, y)$	$f_2(x, y)$	Область
3	$x^2 + \operatorname{tg}^2(y)$	$\ln(\sqrt{x^4 + 2y^2})$	
4	$\sqrt{ x \sin(y) }$	$\ln\left(\frac{1}{x^2} + \frac{1}{y^2}\right)$	
5	$e^{2\sin(x) + \cos(y)}$	$e^{\left(\frac{1}{x} + \frac{1}{y}\right)}$	
6	$x + \sin(y)$	$\operatorname{tg}(x^2 + y^2)$	

Продолжение табл. 2

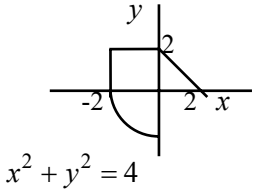
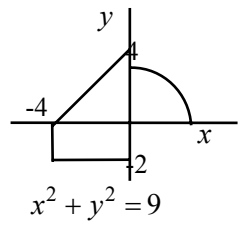
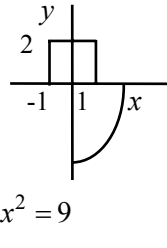
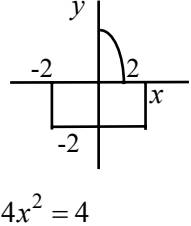
№ п/п	$f_1(x, y)$	$f_2(x, y)$	Область
7	$2x + \operatorname{arctg}(y)$	$\operatorname{arctg}\left\{\frac{1}{x^2} + \frac{1}{y^2}\right\}$	 $x^2 + y^2 = 9$
8	$\ln x^2 + 18y $	$\sqrt{\sin^2(x) + \cos^2(y)}$	 $x^2 + y^2 = 9$
9	$x + \operatorname{tg}(y)$	$\ln x^3 + 3xy $	 $x = y^2/2 - 1$
10	$\operatorname{tg}\left(\frac{y}{\sin^2(x)}\right)$	$\sqrt{\frac{1}{x^2} + \frac{1}{y^2}}$	 $y = x^2 - 2$



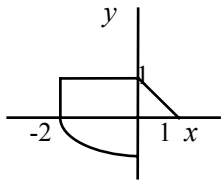
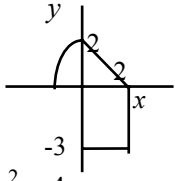
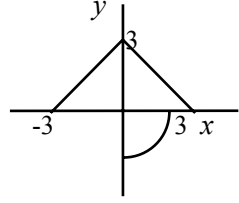
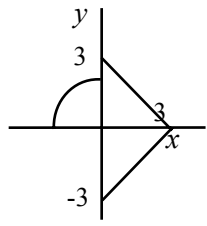
Продолжение табл. 2

№ п/п	$f_1(x, y)$	$f_2(x, y)$	Область
11	$\frac{1}{\sin(x)} + \ln y $	$x^3 + x^2y + y^3$	 $x^2 + y^2 = 1$
12	$\sqrt{x^2 + y^2}$	$x^2 - 10\sin(xy)$	 $x^2 + y^2 = 1$
13	$\sqrt{x^2 + 10y}$	$\sqrt{x^2 + e^y}$	 $x^2 + y^2 = 1$
14	$e^{\cos(y)+x}$	$\sqrt{ \sin(x) + \ln y  }$	 $x + \frac{4}{9}y^2 = 4$

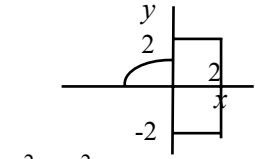
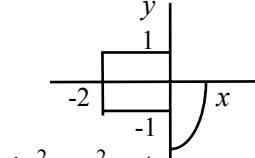
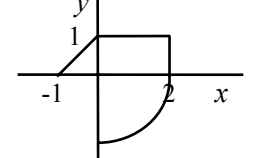
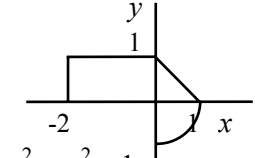
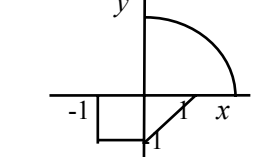
Продолжение табл. 2

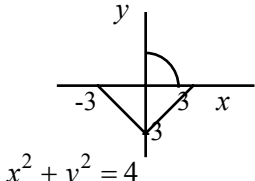
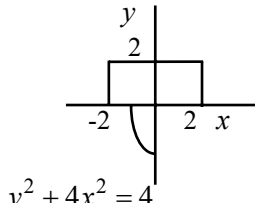
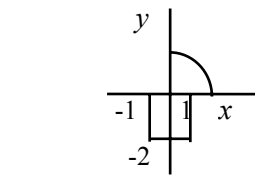
№ п/п	$f_1(x, y)$	$f_2(x, y)$	Область
15	$2\sin(x)\cos(y)$	$\sqrt{ x^2 + xy + y^2 }$	 $x^2 + y^2 = 4$
16	$\frac{\operatorname{tg}(x)}{e^y}$	$\sqrt[3]{\sin^2(2x + 3y)}$	 $x^2 + y^2 = 9$
17	$e^{5x} + 18\cos(y)$	$\frac{\sin(x^2 + y^2)}{x^2 + y^2}$	 $y^2 + \frac{9}{4}x^2 = 9$
18	$\ln\left(\frac{y}{\sin^2(x)}\right)$	$3\ln(x^2) + 5y$	 $y^2 + 4x^2 = 4$

Продолжение табл. 2

№ п/п	$f_1(x, y)$	$f_2(x, y)$	Область
19	$(\sin(x) + \operatorname{tg}(y))^3$	$\sqrt{e^{1/x^2 + 1/y^2}}$	 $4y^2 + x^2 = 4$
20	$\frac{\operatorname{tg}(y)}{1 - \cos^2(x)}$	$\frac{\sin(x)}{\sqrt{ xy }}$	 $4x^2 + y^2 = 4$
21	$\sin^2(x) + \ln y $	$\frac{x + \sin(y)}{\cos(xy)}$	 $x^2 + y^2 = 4$
22	$\ln y - 3\sin(x) $	$\operatorname{arctg}(e^x + y^2)$	 $x^2 + y^2 = 4$

Продолжение табл. 2

№ п/п	$f_1(x, y)$	$f_2(x, y)$	Область
23	$\sqrt{ y } - \cos^2(x)$	$\operatorname{arctg}(e^y + x^2)$	 $y^2 + x^2 / 4 = 1$
24	$\frac{\sin(x)\cos(y)}{1 - \sin^2(x)}$	$\frac{\sin^2(x) + \cos^2(y)}{\sqrt{x^2 + y^2}}$	 $4x^2 + y^2 = 4$
25	$e^{x+\operatorname{tg}(y)}$	$\sqrt{e^{x+\sin(y)}}$	 $y^2 + x^2 = 4$
26	$\cos\left(\frac{ x  +  y }{xy}\right)$	$\frac{\cos^2(x)}{\sin(2x + 3y)}$	 $x^2 + y^2 = 1$
27	$\ln \operatorname{arctg}(y/x) $	$\operatorname{arctg}\left(\frac{1}{x^2} + \frac{1}{y^2}\right)$	 $x^2 + y^2 = 4$

№ п/п	$f_1(x, y)$	$f_2(x, y)$	Область
28	$e^{(5x+8y)/xy}$	$\sqrt{\ln \operatorname{tg}(x) + y }$	 $x^2 + y^2 = 4$
29	$\sqrt{\operatorname{arctg}(y/x)}$	$\sqrt{\sqrt{x^2 + y^2} + y^2}$	 $y^2 + 4x^2 = 4$
30	$\ln y  + e^{\sin(x)}$	$\sqrt{e^{x^2 + y^2}}$	 $x^2 + y^2 = 4$

## ЗАДАЧИ К ЗАЩИТЕ ЛАБОРАТОРНОЙ РАБОТЫ № 2

1. Даны числа  $a, b, c$ . Найти максимальное.
2. Даны числа  $a, b, c$ . Найти минимальное.
3. Даны числа  $a, b, c$ . Найти среднее.
4. Даны числа  $a, b, c$ . Выбрать из них те, которые принадлежат интервалу  $(d, l)$ .
5. Даны числа  $a, b, c$ . Каждое число заменить наибольшим, если числа образуют возрастающую последовательность. В остальных случаях числа оставить без изменений.

6. Даны числа  $a, b, c$ . Каждое число заменить наименьшим, если числа образуют убывающую последовательность. В остальных случаях числа оставить без изменений.
7. Даны числа  $a, b, c$ . Существует ли треугольник с длинами сторон  $a, b, c$ ?
8. Даны числа  $a, b, c$ . Выяснить, имеет ли квадратное уравнение  $ax^2 + bx + c = 0$  действительные корни. Если имеет, то найти их.
9. Даны числа  $a, b, c$ . Исследовать биквадратное уравнение  $ax^4 + bx^2 + c = 0$  на наличие действительных корней и найти их.
10. Выяснить, можно ли прямоугольник со сторонами  $a, b$  уместить внутри прямоугольника со сторонами  $c, d$ .
11. Дать ответ на вопрос: пройдет ли кирпич с ребрами  $a, b, c$  в прямоугольное отверстие со сторонами  $x, y$ .
12. Определить, является ли треугольник с координатами вершин  $(x_1, y_1), (x_2, y_2), (x_3, y_3)$  прямоугольным.
13. Выяснить, является ли четырехугольник с координатами вершин  $(x_1, y_1), (x_2, y_2), (x_3, y_3), (x_4, y_4)$  квадратом.
14. Выяснить, является ли четырехугольник с координатами вершин  $(x_1, y_1), (x_2, y_2), (x_3, y_3), (x_4, y_4)$  ромбом.
15. Выяснить, является ли четырехугольник с координатами вершин  $(x_1, y_1), (x_2, y_2), (x_3, y_3), (x_4, y_4)$  прямоугольником.
16. Даны три числа  $x, y, z$ . Если их сумма меньше 1, то наименьшее заменить полусуммой двух других, иначе наименьшее заменить наибольшим.
17. Даны три числа  $x, y, z$ . Если их сумма больше 1, то наибольшее из  $x$  и  $y$  заменить полусуммой оставшихся, иначе наибольшее из трех чисел заменить наименьшим.
18. Можно ли построить четырехугольник со сторонами  $a, b, c, d$ ?
19. Для заданных значений  $a, b, c$  найти  $z = \max(\min(a, b), c)$ .
20. Даны четыре неравных числа  $a, b, c, d$ . Определить среднее значение двух чисел, которые не являются минимумом и максимумом.
21. Заданы три числа  $x, y, z$ . Если  $x < 0$ , то  $p$  задать как максимальное из  $y$  и  $z$ .
22. Заданы три числа  $x, y, z$ . Если  $z > 0$ , то  $p$  задать как минимальное из  $x$  и  $y$ .
23. Произвольный выпуклый четырехугольник задан координатами своих вершин на плоскости. Есть ли в этом четырехугольнике прямые углы?

24. Заданы три числа  $a, b, c$ . Если  $a < 0$ , то определить  $p$  как минимальное из  $b$  и  $c$ , иначе вычислить  $p$  как среднее арифметическое из  $b$  и  $c$ .

25. Заданы три числа  $a, b, c$ . Если  $a > 0$ , то  $p$  задать как максимальное из  $b$  и  $c$ , иначе вычислить  $p$  как среднее арифметическое из  $a, b$  и  $c$ .

26. Заданы три числа  $a, b, c$ . Если  $a = 0$ , то вычислить  $p$  как среднее арифметическое от  $b$  и  $c$ , иначе минимальное из трех чисел заменить максимальным.

27. Даны две окружности радиусом  $R_1$  и  $R_2$  с центром в начале координат. Определить, где находится точка с координатами  $(x, y)$ : внутри меньшей окружности, в кольце между окружностями или вне их.

28. Заданы координаты двух точек. Определить, лежат ли они на одной окружности с центром в начале координат.

29. Заданы площади круга  $S_1$  и квадрата  $S_2$ . Выяснить, поместится ли круг в квадрате.

30. Заданы площади круга  $S_1$  и квадрата  $S_2$ . Выяснить, поместится ли квадрат в круге.

### ЛАБОРАТОРНАЯ РАБОТА № 3

#### ПРОГРАММИРОВАНИЕ ЦИКЛИЧЕСКИХ ВЫЧИСЛИТЕЛЬНЫХ ПРОЦЕССОВ

**Цель работы.** Получить навыки программирования циклических вычислительных процессов. Изучить циклы с шагом и циклы с условием.

#### ВВЕДЕНИЕ

Работа содержит задачи, которые сводятся к нахождению конечной суммы ряда  $S = \sum a_n(x)$  при различных значениях  $x$ . Каждое слагаемое  $a_n(x)$  зависит от величины  $x$  и номера  $n$ , определяющего место этого слагаемого в сумме. Ряды подобраны таким образом, чтобы при бесконечно большом числе членов ряда их сумма в указанном диапазоне изменения величины  $x$  была конечным числом, зависящим только от  $x$ , т.е. ряды являются сходящимися. Например,

$$S = \sum_{n=0}^{\infty} a_n(x) = \sum_{n=0}^{\infty} (-1)^n x^{2n+1} / (2n+1)! = \sin(x).$$

Первый член этого ряда при  $n=0$  равен  $x$ , второй член при  $n=1$  равен  $(-x^3)/3!$ , третий член при  $n=2$  равен  $x^5/5!$  и т.д. В сумме при бесконечно большом числе членов они дают функцию  $\sin(x)$ . Видно, что при ограниченных значениях  $x$ , сравнимых по модулю с единицей, каждый последующий член этого знакопеременного ряда существенно меньше предыдущего, поэтому ряд быстро сходится и для вычисления суммы ряда с необходимой точностью достаточно просуммировать относительно небольшое число членов ряда. Например, суммирование можно прекратить, когда текущий член ряда станет по абсолютной величине меньше заданного малого положительного числа  $\epsilon$ , определяющего погрешность вычисления суммы ряда.

Формула общего члена суммы принадлежит к одному из следующих типов:

а)  $x^n/n!$ ;  $(-1)^n x^{2n-1}/(2n-1)!$ ;  $x^{2n}/(2n)!$ ;

б)  $\cos(nx)/n$ ;  $\sin((2n-1)x)/(2n-1)$ ;  $\cos(2nx)/(4n^2-1)$ ;

в)  $x^{4n+1}/(4n+1)$ ;  $(x/2)^2(n^2+1)/n!$ .

В случае а) для вычисления члена суммы целесообразно использовать рекуррентные соотношения, т.е. выражать последующий член суммы через предыдущий, например, если

$$a_n = x^n/n!, \text{ то}$$

$$a_{n+1} = x^{n+1}/(n+1)! = (x^n/n!)(x/(n+1)) = a_n x/(n+1).$$

В случае б) каждый член суммы вычисляется согласно общей формуле, которую и нужно запрограммировать.

В случае в) член суммы целесообразно представить в виде двух переменных, одна из которых вычисляется по рекуррентному соотношению, а другая – непосредственно, например, если

$$a_n = (x/2)^2(n^2+1)/n! \text{ то}$$

$$a_n = c_n/b_n, \text{ где } b_n = b_{n-1}n, c_n = (x/2)^2(n^2+1).$$



### ЗАДАНИЕ

1. Записать расчетные формулы для вычисления слагаемого и конечной суммы ряда  $S = \sum a_n(x)$ . Варианты заданий приведены в табл. 3.

2. Составить алгоритм вычисления с заданной точностью суммы конечного числа членов ряда в указанном диапазоне изменения параметра  $x$  с заданным шагом. Для получения шага диапазон изменения  $x$  разделить на 10.

3. Составить программу решения задачи. В программе для каждого значения  $x$  предусмотреть вычисление как приближенной, так и точной суммы соответствующего ряда  $y$  по приведенной в табл. 3 формуле.

4. Отладить программу. Результат получить в виде таблицы, содержащей для каждого  $x$  из указанного диапазона приближенное и точное значения суммы ряда, а также количество просуммированных членов.

### ПРИМЕР ВЫПОЛНЕНИЯ ЗАДАНИЯ

Пусть требуется составить программу для вычисления суммы ряда:

$$S = -(2x)^2/2 + (2x)^4/24 + \dots + (-1)^n (2x)^{2n}/(2n)! + \dots$$

и функции  $y = 2(\cos^2(x) - 1)$  для  $0.1 \leq x \leq 1.0$ . Суммирование следует выполнять до тех пор, пока текущий член ряда по абсолютной величине не станет меньше заданного малого положительного числа  $\epsilon$ .

Запишем ряд в виде

$$S = \sum a_n,$$

где  $a_n = (-1)^n (2x)^{2n}/(2n)! = c_n b_n$ .

Здесь  $c_n = (-1)^n$ ;  $b_n = (2x)^{2n}/(2n)!$

Формула общего члена ряда относится к типу а), т.е. для вычисления члена суммы целесообразно использовать рекуррентные соотно-

шения, выражая каждый последующий член суммы через предыдущий. Суммирование начинается с  $n = 1$ , при этом начальные значения  $c = -1$ ,  $b = 1$ .

Для каждого значения  $x$ , задаваемого во внешнем цикле, вычисление суммы ряда выполняется во внутреннем цикле, при прохождении которого номер члена ряда увеличивается на 1, а сумма изменяется по формуле  $S = S + a$ , где  $a$  вычисляется как произведение  $c$  и  $b$ . Начальное значение суммы в данном примере равно нулю  $S = 0$ . Суммирование в цикле DO WHILE продолжается до тех пор, пока величина  $b$  по абсолютной величине больше  $\varepsilon$ . Вычисление постоянной по модулю знакопеременной величины  $c_n$  можно организовать в цикле по формуле  $c = -c$ . Для вычисления величины  $(2n)!$  в знаменателе необходимо во внутреннем цикле, где  $n$  каждый раз увеличивается на 1, число  $2n$  умножать на число  $(2n - 1)$ , так как факториал – это по определению произведение последовательных натуральных чисел.

### Фортран–программа

```

PROGRAM LAB3
REAL A, B, C, EPS, H, S, X, X0, XK, Y
INTEGER I, N
OPEN (1, FILE='LAB3.TXT')
X0=0.1; XK=1.; EPS=0.0001
H=(XK-X0)/10.
DO I=0,10
  X=X0+H*I
  N=1; C=-1.; B=1.; S=0.
  DO WHILE(ABS(B) > EPS)
    B=B*(2.*X)**2/(2.*N*(2.*N-1.)); A=C*B
    N=N+1; S=S+A; C=-C;
  END DO
  Y=2.*(COS(X)**2-1.)
  PRINT 2, X,S,Y,N; WRITE (1,2) X,S,Y,N
END DO
2 FORMAT (' X=',F6.3,3X,'S=',E10.3,3X,'Y=',E10.3,3X,'N=',I5)
END PROGRAM LAB3

```

Результаты счета выводятся не только на дисплей, но и в файл LAB3.TXT. Вывод осуществляется под управлением оператора FORMAT с меткой 2, который формирует таблицу, содержащую для

каждого  $x$  из указанного диапазона приближенное и точное значения суммы ряда, а также количество просуммированных членов.

Таблица 3

**Варианты заданий к лабораторной работе № 3**

№ п/п	Формула ряда	$x_0 \leq x \leq x_k$		Точное значение суммы ряда $y$
		$x_0$	$x_k$	
1	$S = 1 + x(\ln 3)/1 + x^2(\ln^2 3)/2! + \dots$ $+ x^n(\ln^n 3)/n! + \dots$	0.1	1.0	$3^x$
2	$S = \cos x + (\cos 2x)/2 + \dots$ $+ (\cos nx)/n + \dots$	$\pi/5$	$9\pi/5$	$-\ln(2 \sin(x/2))$
3	$S = x - x^3/3! + \dots$ $+ (-1)^n x^{2n+1}/(2n+1)! + \dots$	0.1	1.0	$\sin x$
4	$S = \sin x - (\sin 2x)/2 + \dots$ $+ (-1)^{n+1}(\sin nx)/n + \dots$	$\pi/5$	$4\pi/5$	$x/2$
5	$S = 1 + x/1 + x^2/2! + \dots + x^n/n! + \dots$	1.0	2.0	$e^x$
6	$S = 1 + x \cos(\pi/4) + \dots$ $+ x^n \cos(n\pi/4)/n! + \dots$	0.1	1.0	$\cos(x \sin(\pi/4)) \cdot e^{x \cos(\pi/4)}$
7	$S = 1 - x^2/2! + \dots$ $+ (-1)^n x^{2n}/(2n)! + \dots$	0.1	1.0	$\cos x$
8	$S = x \sin(\pi/4) + \dots$ $+ x^n \sin(n\pi/4) + \dots$	0.1	0.8	$\frac{x \sin(\pi/4)}{1 - 2x \cos(\pi/4) + x^2}$
9	$S = x + x^5/5 + \dots$ $+ x^{4n+1}/(4n+1) + \dots$	0.1	0.8	$\frac{1}{4} \ln \frac{1+x}{1-x} + \frac{1}{2} \operatorname{arctg} x$
10	$S = 1 + \cos x + \dots + \cos(nx)/n! + \dots$	0.1	1.0	$\cos(\sin x) e^{\cos x}$
11	$S = 1 + 3x^2 + \dots + (2n+1)x^{2n}/n! + \dots$	0.1	1.0	$(1 + 2x^2) e^{x^2}$
12	$S = x \cos(\pi/3) + x^2 \cos(2\pi/3)/2 + \dots$ $+ \dots + x^n \cos(n\pi/3)/n + \dots$	0.1	0.8	$\frac{-\ln(x^2 - 2x \cos(\pi/3) + 1)}{2}$

Продолжение табл. 3

№ п/п	Формула ряда	$x_0 \leq x \leq x_k$		Точное значение суммы ряда $y$
		$x_0$	$x_k$	
13	$S = \frac{x-1}{x+1} + \frac{1}{3} \left( \frac{x-1}{x+1} \right)^3 + \dots$ $+ \frac{1}{2n+1} \left( \frac{x-1}{x+1} \right)^{2n+1} + \dots$	0.2	1.0	$\frac{\ln x}{2}$
14	$S = -\cos x + \frac{\cos 2x}{4} + \dots$ $+ (-1)^n \frac{\cos nx}{n^2} + \dots$	$\pi/5$	$\pi$	$\frac{1}{4} (x^2 - \pi^2/3)$
15	$S = x^3/3 - x^5/15 + \dots$ $+ (-1)^{n+1} x^{2n+1} / (4n^2 - 1) + \dots$	0.1	1.0	$\frac{1+x^2}{2} \operatorname{arctg} x - \frac{x}{2}$
16	$S = \sin x + \frac{1}{3} \sin 3x + \dots$ $+ \frac{1}{2n-1} \sin((2n-1)x) + \dots$	$\pi/10$	$9\pi/10$	$\pi/4$
17	$S = 1 + x^2/2! + \dots + x^{2n}/(2n)! + \dots$	0.1	1.0	$\frac{1}{2} (e^x + e^{-x})$
18	$S = \frac{1}{3} \cos 2x + \frac{1}{15} \cos 4x + \dots$ $+ \frac{1}{4n^2 - 1} \cos(2nx) + \dots$	0.1	0.8	$\frac{1}{2} - \frac{\pi}{4}  \sin x $
19	$S = 1 + 2x + \dots + (2x)^n / n! + \dots$	0.1	1.0	$e^{2x}$
20	$S = 1 + 2\frac{x}{2} + \dots + \frac{n^2+1}{n!} \left( \frac{x}{2} \right)^n + \dots$	0.1	1.0	$\left( 1 + x/2 + x^2/4 \right) e^{x/2}$

Продолжение табл. 3

№ п/п	Формула ряда	$x_0 \leq x \leq x_k$		Точное значение суммы ряда $y$
		$x_0$	$x_k$	
21	$S = x - x^3/3 + \dots + (-1)^n \frac{x^{2n+1}}{2n+1} + \dots$	0.1	0.5	$\operatorname{arctg} x$
22	$S = 1 - \frac{3}{2}x^2 + \dots$ $+ (-1)^n \frac{2n^2+1}{(2n)!} x^{2n} + \dots$	0.1	1.0	$\left(1 - \frac{x^2}{2}\right) \cos x - \frac{x}{2} \sin x$
23	$S = -\frac{(2x)^2}{2} + \frac{(2x)^4}{24} + \dots$ $+ (-1)^n \frac{(2x)^{2n}}{(2n)!} + \dots$	0.1	1.0	$2(\cos^2(x) - 1)$
24	$S = -(1+x)^2 + \frac{(1+x)^4}{2} + \dots$ $+ (-1)^n \frac{(1+x)^{2n}}{n} + \dots$	-1.9	-1.1	$\ln \frac{1}{2+2x+x^2}$
25	$S = x + \frac{x^3}{3!} + \dots + \frac{x^{2n+1}}{(2n+1)!} + \dots$	0.1	1.0	$\frac{1}{2}(e^x - e^{-x})$
26	$S = x/3! + 4x^2/5! + \dots$ $+ n^2 x^n / (2n+1)! + \dots$	0.2	0.8	$\frac{1}{4} \left( \frac{x+1}{\sqrt{x}} \operatorname{sh}(\sqrt{x}) - \operatorname{ch}(\sqrt{x}) \right)$
27	$S = x \cos(\pi/4) + x^2 \cos(2\pi/4) + \dots$ $+ x^n \cos(n\pi/4) + \dots$	0.1	0.8	$\frac{x \cos(\pi/4) - x^2}{1 - 2x \cos(\pi/4) + x^2}$
28	$S = 3x + 8x^2 + \dots + n(n+2)x^n + \dots$	0.1	0.8	$x(3-x)/(1-x)^3$

№ п/п	Формула ряда	$x_0 \leq x \leq x_k$		Точное значение суммы ряда $y$
		$x_0$	$x_k$	
29	$S = \cos x + \frac{\cos(3x)}{9} + \dots$ $+ \frac{\cos((2n-1)x)}{(2n-1)^2} + \dots$	$\pi/5$	$\pi$	$\pi^2/8 - (\pi/4) x $
30	$S = x^2/2 - x^4/12 + \dots$ $+ (-1)^{n+1} \frac{x^{2n}}{2n(2n-1)} + \dots$	0.1	0.8	$x \operatorname{arctg} x - \ln \sqrt{1+x^2}$

**ЗАДАЧИ К ЗАЩИТЕ ЛАБОРАТОРНОЙ РАБОТЫ №3**

1. Получить таблицу температур по Цельсию от 0 до 100 градусов с шагом 10 градусов и эквивалентных температур по шкале Фаренгейта, используя для перевода формулу  $T_f = 32 + \frac{9}{5}T_c$ .

2. Вычислить последовательности значений функций  $P_1(x) = x$ ,  $P_2(x) = (3x^2 - 1)/2$ ,  $P_3(x) = (5x^2 - 3x)/2$  для значений аргумента  $x = 1, 2, \dots, 20$ .

3. На отрезке  $[a, b]$  с шагом  $h = (b-a)/n$ , где  $n$  – натуральное число, вычислить значения функции  $y = \sin x + \cos 2x$  при  $a = -\pi$ ,  $b = \pi$ ,  $n = 20$ .

4. На отрезке  $[a, b]$  с шагом  $h = (b-a)/n$ , где  $n$  – натуральное число, вычислить значения функции  $y = \sin \sqrt{(2x + \cos x)}$  при  $a = 0$ ,  $b = 2\pi$ ,  $n = 15$ .

5. На отрезке  $[a, b]$  с шагом  $h = (b-a)/n$ , где  $n$  – натуральное число, вычислить значения функции  $y = |x|e^{-x}$  при  $a = -1$ ,  $b = 3$ ,  $n = 20$ .

6. Даны натуральные числа  $m$  и  $n$ . Получить  $\frac{m! + n!}{(m+n)!}$ .

7. Вычислить  $\prod_{i=1}^{40} \frac{i^2}{i^2 + 2i + 3}$ .
8. Вычислить  $\prod_{i=1}^{10} \left(2 + \frac{1}{i!}\right)$ .
9. Вычислить  $\prod_{i=1}^{50} \frac{i+1}{i+2}$ .
10. Вычислить  $\prod_{i=2}^{10} \left(1 - \frac{1}{i!}\right)^2$ .
11. Даны натуральное число  $n$  и действительное  $x$ . Вычислить  $\prod_{k=1}^n \left(1 + \frac{\sin kx}{k!}\right)$ .
12. Даны натуральное число  $n$  и действительное  $x$ . Вычислить  $\prod_{k=1}^n \left(\frac{k}{k+1} - \cos^k(x)\right)$ .
13. Дано действительное число  $a > 0$ . Последовательность  $x_1, x_2, \dots$  образована по закону  $x_n = \frac{4}{5}x_{n-1} + \frac{a}{5}x_{n-1}$ , где  $x_0 = \min(2a, 5)$ . Найти значение  $n$ , для которого  $|x_n - x_{n-1}| < 10^{-5}$ . Вычислить разность  $a - x_n^3$ .
14. Даны действительные положительные числа  $a, x$ . В последовательности  $y_1, y_2, \dots$ , образованной по закону  $y_i = \frac{1}{2}(y_{i-1} + x/y_{i-1})$ , где  $y_0 = a$ , найти первый член  $y_n$ , для которого  $|y_n^2 - y_{n-1}^2| < 10^{-4}$ .
15. Пусть  $x_0 = 1, x_k = (2 - x_{k-1}^3)/5$ , где  $k = 1, 2, \dots$ . Найти  $n$ , для которого  $|x_n - x_{n-1}| < 10^{-5}$ .
16. Пусть  $a_0 = a_1 = 1, a_i = a_{i-2} + a_{i-1}/2^{i-1}$ , где  $i = 2, 3, \dots$ . Найти произведение  $a_0 a_1 \dots a_9$ .
17. Дано натуральное число  $n$ . Сколько цифр в числе  $n$ ?

18. Дано натуральное число  $n$ . Чему равна сумма его цифр?
19. Дано натуральное число  $n$ . Найти первую цифру числа  $n$ .
20. Дано натуральное число  $n$ . Получить сумму двух последних цифр числа  $n$ .
21. Дано натуральное число  $n$ . Выяснить, входит ли цифра 3 в запись числа  $n$ .
22. Дано натуральное число  $n$ . Переставить первую и последнюю цифры числа.
23. Дано натуральное число  $n$ . Вычислить  $1/\sin 1 + 1/(\sin 1 + \sin 2) + \dots + 1/(\sin 1 + \dots + \sin n)$ .
24. Дано натуральное число  $n$ . Вычислить  $\cos 1/\sin 1 + (\cos 1 + \cos 2)/(\sin 1 + \sin 2) + \dots + (\cos 1 + \dots + \cos n)/(\sin 1 + \dots + \sin n)$ .
25. Дано действительное число  $a$  и натуральное  $n$ . Вычислить  $a(a+1)\dots(a+n-1)$ .
26. Дано действительное число  $a$  и натуральное  $n$ . Вычислить  $\frac{1}{a} + \frac{1}{a(a+1)} + \dots + \frac{1}{a(a+1)\dots(a+n)}$ .
27. Дано действительное число  $a$  и натуральное  $n$ . Вычислить  $a(a-n)(a-2n)\dots(a-n^2)$ .
28. Дано действительное число  $a$  и натуральное  $n$ . Вычислить  $\sin x + \sin \sin x + \dots + \underbrace{\sin \sin \dots \sin x}_n$ .
29. Дано действительное число  $a$ . Найти среди чисел  $1, 1+1/2, 1+1/2+1/3, \dots$  первое, большее  $a$ .
30. Дано действительное число  $a$ . Найти такое наименьшее  $n$ , что  $1+1/2+\dots+1/n > a$ .

#### ЛАБОРАТОРНАЯ РАБОТА № 4

#### ИСПОЛЬЗОВАНИЕ МАССИВОВ

**Цель работы.** Изучить особенности описания и использования массивов в программе.



## ЗАДАНИЕ

Разработать алгоритм решения выбранной по номеру варианта задачи, составить и отладить программу с использованием массивов.

## ВВЕДЕНИЕ

Наряду с простыми переменными в алгоритмических языках используются переменные с индексами, называемые массивами. Массив – это упорядоченная последовательность величин, обозначаемая одним именем. Величины, из которых состоит массив, называются элементами массива. Элемент массива обозначается с помощью имени массива и индексов, следующих в круглых скобках за именем массива. Индексы отделяются друг от друга запятыми и указывают положение элемента в массиве. Индексов может быть 1, 2 и так далее вплоть до 7. Соответственно массивы бывают одномерные, двумерные и т.д.

Все массивы, используемые в программе, должны быть описаны, так как для хранения элементов массива в определенной последовательности отводится необходимый объем памяти. При описании указываются имя массива и затем в круглых скобках через запятую – граничные пары, т.е. разделенные двоеточием минимальные и максимальные значения соответствующего индекса. При отсутствии двоеточия минимальное значение соответствующего индекса в Фортране по умолчанию равно единице. Для описания массивов в Фортране используются операторы описания типа переменных (REAL, INTEGER, COMPLEX, LOGICAL, CHARACTER, DOUBLE PRECISION), а также атрибут и оператор DIMENSION.

Рассмотрим несколько примеров описания массивов. Значения граничных пар можно задавать с использованием именованных констант и константных выражений.

```
INTEGER, PARAMETER :: N=4
```

```
REAL, DIMENSION :: A(N,N), B(N), K(N+1,N+1), C(0:N)
```

В этом примере оператор REAL с атрибутом DIMENSION описывает двумерный вещественный массив A, содержащий 16 элементов; одномерный вещественный массив B, содержащий 4 элемента; двумерный вещественный массив K, содержащий 25 элементов, и одномерный вещественный массив C, содержащий 5 элементов.

Объекты с общими атрибутами можно группировать в единый список, например, следующий оператор описывает два двумерных веще-

ственных массива D и E, каждый из которых имеет 3 строки и 4 столбца.

```
REAL, DIMENSION (3,4) :: D, E
```

Операторы описания типа могут использоваться для описания массивов и без атрибутов, но с указанием граничных пар, например,

```
INTEGER MIN(2,2), MAX(10)  
REAL Q(5), SUMMA(3,3)
```

Здесь MIN и MAX – целые массивы, содержащие 4 и 10 элементов соответственно; Q и SUMMA – вещественные массивы, содержащие 5 и 9 элементов.

Оператор DIMENSION также может использоваться для описания массивов, например, для описания одномерного вещественного массива F из 10 элементов

```
REAL F  
DIMENSION F(-2:7)
```

Использование оператора DIMENSION без операторов описания типа, когда тип массива определяется по умолчанию возможно, но нежелательно.

**Операторы, описывающие массивы, необходимо располагать в начале программы перед первым исполняемым оператором.**

В Фортране список операторов ввода READ и вывода PRINT или WRITE может содержать как имена простых переменных и элементов массива, так и имена массивов. В последнем случае вводу-выводу подлежат все элементы массива в «естественной последовательности» (т.е. сначала растет первый индекс, затем второй и т.д.). Так, если в программе описан массив A(2,2) и имеется оператор ввода READ \*, A, то численные значения необходимо задавать в такой последовательности: A(1,1), A(2,1), A(1,2), A(2,2), т.е. любая матрица вводится по столбцам. Для изменения порядка ввода-вывода или если требуется передать только часть элементов массива, можно использовать неявную форму оператора цикла. Например, ввод матрицы A(2,2) по строкам осуществляет оператор READ \*, ((A(I,J), J=1,2), I=1,2).

Не следует забывать, что в Фортране функции «что вводить (выводить)» и «как вводить (выводить)» распределены между операторами ввода-вывода и оператором FORMAT.

Если в программе предусмотрен ввод исходных данных с клавиатуры, то в процессе отладки приходится задавать одни и те же численные значения много раз. Поэтому, когда исходных данных достаточно много,

удобнее записать их перед счетом программы в файл данных, а в программе предусмотреть оператор, считывающий данные из файла данных. Кроме того, результаты также имеет смысл выводить в файл данных, чтобы иметь возможность получить распечатку результатов счета. Обращение к файлу данных производится по его логическому номеру N, указанному в операторах READ (N,...) или WRITE (N,...). С помощью оператора OPEN можно как назначать логические номера уже имеющимся файлам, так и создавать новые файлы с указанием соответствующего логического номера, например, OPEN (1, FILE='LAB4.TXT'). После выполнения этого оператора файлу LAB4.TXT будет соответствовать логический номер 1. Файл LAB4.TXT создается в папке проекта либо предварительно до запуска программы на счет, либо в процессе работы программы в момент выполнения оператора OPEN. Для завершения доступа к файлу используется оператор CLOSE.

### **ТИПОВЫЕ ПРИМЕРЫ ИСПОЛЬЗОВАНИЯ МАССИВОВ В ПРОГРАММАХ**

Отличительной особенностью Фортрана является развитая система операций с массивами. Операции над массивами и сечениями массивов задают параллелизм действий над компонентами массивов (или массива). Такие средства, с одной стороны, позволяют пользователю лаконично и сжато описать алгоритм обработки массивов и, с другой стороны, дают возможность компилятору генерировать более эффективный код с учетом особенностей конкретного компьютера. Очевидно, что наиболее эффективно использование этих возможностей для вычислительных систем, имеющих аппаратные средства для векторной обработки.

В Фортран встроено большое число функций, позволяющих:

- выполнять вычисления в массивах, например, суммировать элементы массива или находить его максимальный элемент;
- преобразовывать массивы, например, можно получать из одномерного массива двумерный;
- получать справочные данные о массиве (размер, форма и значения границ каждого измерения).

1. Ввод одномерного массива B(8) из файла данных:

READ (1,\*) B

Численные значения (8 значений) перед запуском программы на счет записываются через запятую или пробел в файл данных, соответствующий логическому номеру 1.

2. Ввод двумерного массива A(4,4) из файла данных:

а) построчно с использованием неявной формы цикла

```
READ (1,*) ((A(I,J), J=1,4), I=1,4)
```

Перед выполнением программы численные значения записываются в файл данных, соответствующий логическому номеру 1, в четыре строки по четыре элемента в строку через запятую или пробел;

б) построчно с использованием сечения массива (простейшей формы индексного триплета)

```
READ (1,*) (A(I,:),I=1,4)
```

Файл данных такой же, как и в случае а).

Индексный триплет применяется для получения доступа к некоторому подмножеству элементов массива, называемому сечением массива. При задании сечения массива индексный триплет подставляется вместо одного из индексов массива.

Индексный триплет имеет вид:

[нижняя граница]:[верхняя граница] [:шаг]

Все параметры триплета являются необязательными. При отсутствии нижней (верхней) границы триплета ее значение принимается равным значению нижней (верхней) границы соответствующего индекса массива. При отсутствии шага его значение принимается равным 1.

3. Вывод одномерного массива B(4) на экран:

```
PRINT 5, B
```

```
5      FORMAT (' VECTOR B'/4F10.4)
```

При выводе в файл данных оператор PRINT заменяется на оператор WRITE, например, при выводе двумерного массива A(4,4)

```
WRITE (1,10) (A(I,:), I=1,4)
```

```
10     FORMAT (' MATRICA A'/4(4F10.4/))
```

4. Суммирование элементов одномерного массива B(10)  $S = \sum_{i=1}^{10} b_i$  :

а) с использованием встроенной функции SUM

```
S=SUM(B),
```

б) по следующему алгоритму: начальное значение суммы принимается равным нулю  $S = 0$ , затем в цикле по  $i$  от 1 до 10 накапливается сумма  $S = S + b_i$

```

S=0.
DO I=1,10
  S=S+B(I)
END DO.

```

5. Произведение элементов одномерного массива B(10)  $P = \prod_{i=1}^{10} b_i$  :

а) целесообразно использовать встроенную функцию PRODUCT

```

P=PRODUCT(B),

```

б) можно составить алгоритм, аналогичный алгоритму сложения элементов массива, в котором начальное значение произведения необходимо положить равным единице, а операцию сложения заменить произведением

```

P=1.
DO I=1,10
  P=P*B(I)
END DO.

```

6. Пусть требуется определить максимальный элемент массива B из N элементов и его номер K:

а) Определение максимального (или минимального) элемента одномерного массива  $S = \max(b_1, b_2, \dots, b_n)$  и номера  $k$  этого элемента выполняется по следующему алгоритму: сначала за максимальный принимается первый элемент массива  $S = b_1$  и, следовательно, номером  $k$  максимального элемента будет единица  $k = 1$ . Затем в цикле по  $i$  от 2 до N текущее значение максимального  $S$  сравнивается с  $b_i$ . Если условие  $S < b_i$  не выполняется, то цикл повторяется. Если условие  $S < b_i$  выполняется, то текущему значению максимального  $S$  присваивается значение  $b_i$ , а его номеру – значение  $i$ .

```

INTEGER, PARAMETER :: N=10
REAL B(N) /1.,2.,3.,4.,5.,6.,5.,4.,3.,2./
S=B(1); K=1
DO I=2,N
  IF (S<B(I)) THEN
    S=B(I); K=I
  END IF
END DO

```

```
PRINT *, ' S=', S, ' K=', K
END,
```

б) Определение максимального (или минимального) элемента одномерного массива  $S = \max(b_1, b_2, \dots, b_n)$  и номера  $k$  этого элемента можно выполнить с использованием встроенных функций MAXVAL и MAXLOC (MINVAL, MINLOC).

```
INTEGER, PARAMETER :: N=10
REAL B(N) /1.,2.,3.,4.,5.,6.,5.,4.,3.,2./
INTEGER K(1)
S=MAXVAL(B); K=MAXLOC(B)
PRINT *, ' S=', S, ' K=', K
END.
```

7. Пусть требуется расположить элементы одномерного массива B(10) в порядке невозрастания, т.е. так, чтобы для всех элементов массива выполнялось условие:  $b_i \geq b_{i+1}$  при  $i=1,9$ . Рассмотрим один из способов решения этой задачи. Сначала из всех элементов  $b_i$  при  $i=1,10$  найти максимальный (его величину  $S$  и номер  $k$ ), затем переставить первый и максимальный элементы. Из оставшихся девяти элементов  $b_i$  при  $i=2,10$  найти максимальный и поменять его местами со вторым и т.д. Перестановка двух элементов массива с номерами  $k$  и  $j$  выполняется следующим образом: элемент  $b_k$  равен  $S$ , поэтому сначала следует заменить  $b_k$  на  $b_j$ , а затем  $b_j$  присваивается значение  $S$ . Программа упорядочения элементов массива будет иметь следующий вид:

```
DO J=1,9
  S=B(J); K=J
  DO I=J+1,10
    IF (S<B(I)) THEN
      S=B(I); K=I
    END IF
  END DO
  B(K)=B(J); B(J)=S
END DO.
```

Внешний цикл по  $j$  изменяет номер элемента, с которого начинается поиск.

8. Пусть требуется составить одномерный массив C(4) из максимальных элементов строк матрицы A(4,4). Программа строится по схеме двойного цикла. Параметром внешнего цикла является номер строки  $i$ . Во внутреннем цикле по  $j$  определяется максимальный элемент строки:

```
DO I=1,4
  C(I)=A(I,1)
  DO J=2,4
    IF (C(I)<A(I,J)) C(I)=A(I,J)
  END DO
END DO.
```

9. Пусть требуется умножить матрицу A(4,4) на вектор B(4). Результатом умножения матрицы A(4,4) на вектор B(4) является вектор C(4), элементы которого определяются по формуле

$$C_i = \sum_{j=1}^4 a_{ij} b_j$$

а) используем встроенную функцию MATMUL, которая выполняет матричное умножение операндов

C=MATMUL(A,B),

б) алгоритм строится по схеме двойного цикла: внешний цикл по параметру  $i$ , во внутреннем цикле по параметру  $j$  производится суммирование:

```
DO I=1,4
  C(I)=0.
  DO J=1,4
    C(I)=C(I)+A(I,J)*B(J)
  END DO
END DO.
```

10. Пусть требуется найти произведение D матрицы A на матрицу F, где A(4,4), F(4,4) и D(4,4). Умножение матрицы на матрицу осуществляется по схеме тройного цикла в соответствии с формулой

$$d_{ij} = \sum_{k=1}^4 a_{ik} f_{kj}, \text{ где } i=1,4 \text{ и } j=1,4$$

а) используем схему тройного цикла с суммированием во внутреннем цикле

```

DO I=1,4
  DO J=1,4
    D(I,J)=0.
    DO K=1,4
      D(I,J)=D(I,J)+A(I,K)*F(K,J)
    END DO
  END DO
END DO,

```

б) используем схему двойного цикла со встроенной функцией суммирования

```

DO I=1,4
  DO J=1,4
    D(I,J)=SUM(A(I,:)*F(:,J))
  END DO
END DO.

```

11. Пусть требуется вычислить элементы матрицы D(4,4) по формулам:

$$d_{ij} = \begin{cases} (b_i + b_j)^2, & b_i + b_j < -1 \\ 2(b_i + b_j), & -1 \leq b_i + b_j \leq 1 \\ \sqrt{b_i + b_j}, & b_i + b_j > 1 \end{cases}.$$

Программа, использующая блочный оператор IF, имеет следующий вид:

```

DO I=1,4
  DO J=1,4
    Z= B(I)+B(J)
    IF (Z<-1.) THEN
      D(I,J)= Z**2
    ELSE IF (Z>1.) THEN
      D(I,J)=SQRT(Z)
    ELSE
      D(I,J)=2.*Z
    END IF
  END DO
END DO.

```



#### ВАРИАНТЫ ЗАДАНИЙ К ЛАБОРАТОРНОЙ РАБОТЕ № 4

В вариантах заданий в качестве исходных данных используются следующие массивы:

$$A = \begin{pmatrix} -1.06 & 3.29 & -2.93 & 0.46 \\ 1.85 & -4.05 & 1.96 & -2.17 \\ 0.36 & 2.85 & -3.02 & 1.57 \\ 2.87 & -1.83 & 2.89 & -0.93 \end{pmatrix}; \quad B = \begin{pmatrix} 1.07 \\ -2.36 \\ 1.98 \\ 0.76 \end{pmatrix}.$$

1. Заданы массивы A(4,4), B(4). Вычислить элементы массивов C(4), D(4,4), F(4,4) в соответствии с формулами:

$$c_i = \begin{cases} \operatorname{tg}(0.1i + 1), & i = 1, 2 \\ \operatorname{ctg}(0.2i + 1), & i = 3, 4 \end{cases}, \quad d_{ij} = c_i + b_j, \quad f_{ij} = a_{ij} + d_{ij}.$$

Вывести на печать массивы C, D, F.

2. Заданы массивы A(4,4), B(4). Вычислить элементы массива по формуле:

$$c_i = \begin{cases} \sqrt{b_i}, & b_i > 0 \\ b_i^2, & b_i \leq 0 \end{cases}.$$

Найти максимальный элемент массива P=C+B. Преобразовать матрицу A, разделив все ее элементы на найденный максимум. Печатать A, B, C, P.

3. Заданы массивы A(4,4), B(4). Из 1-й строки A вычесть 3-ю и в полученной матрице найти максимальный элемент. Из всех элементов массива B вычесть найденный максимум. Печатать A, B.

4. Задан массив B(4). Вычислить элементы массива C(4) по формулам:

$$c_i = \begin{cases} b_i^2, & b_i < -1 \\ 1 + b_i, & -1 \leq b_i \leq 1 \\ \sqrt{b_i}, & b_i > 1 \end{cases}.$$

Вычислить элементы матрицы  $D(4,4)$  по формуле  $d_{ij} = b_i c_j$ . Найти

$S = \sum_{i=1}^4 b_i c_i$ , из всех элементов массива  $B$  вычесть  $S$ . Печатать  $B, C, D, S$ .

5. Задан массив  $A(4,4)$ . Составить вектор  $C(4)$  из минимальных элементов в столбцах  $A$ . Из каждой строки  $A$  вычесть  $C$ . Печатать  $A, C$ .

6. Задан массив  $A(4,4)$ . Составить вектор  $C(4)$  из максимальных модулей элементов в строках  $A$ . Элементы вектора  $C$  разделить на максимальный элемент  $C$ . Из диагональных элементов  $A$  вычесть  $C$ . Печатать  $A, C$ .

7. Заданы массивы  $A(4,4), B(4)$ . Найти  $C=AB$ . Максимальный элемент  $C$  и его номер  $k$ . Из  $k$ -й строки матрицы  $A$  вычесть  $C$ . Печатать  $A, B, C, k$ .

8. Заданы массивы  $A(4,4), B(4)$ . Найти максимальные элементы массивов  $A$  и  $B$  и поменять их местами. Печатать  $A, B$ .

9. Задан массив  $B(4)$ . Вычислить элементы массива  $C(4)$  по формулам:

$$c_i = \begin{cases} 1 + \sqrt{b_i}, & b_i > 0 \\ 2b_i^2, & b_i \leq 0 \end{cases}.$$

Вычислить элементы матриц  $D(4,4)$  ( $d_{ij} = b_i + c_j$ ) и  $F(4,4)$  ( $f_{ij} = b_i c_j$ ). Найти  $Q=DF$ . Печатать  $C, D, F, Q$ .

10. Заданы массивы  $A(4,4), B(4)$ . Составить вектор  $C(4)$  из максимальных элементов в строках  $A$ . Вычислить  $P=BC$ . Из диагональных элементов  $A$  вычесть  $P$ . Печатать  $A, B, C, P$ .

11. Задан массив  $A(4,4)$ . Вычислить элементы массива по формулам. Найти  $F=AD$ . Печатать  $A, D, F$ .

$$d_{ij} = \begin{cases} \sqrt{a_{ij}}, & a_{ij} > 0 \\ a_{ij}^2 - 1, & a_{ij} \leq 0 \end{cases}.$$

12. Заданы массивы  $A(4,4), B(4)$ . Найти максимальный элемент  $B$  и его номер  $k$ , минимальный элемент  $B$  и его номер  $l$ . В матрице  $A$  элемент  $a_{kl}$  заменить на найденный максимум,  $a_{lk}$  – на минимум. В векторе  $B$  поменять местами максимальный и минимальный элементы. Вычислить  $C=AB$ . Печатать  $A, B, C$ .

13. Заданы массивы A(4,4), B(4). Упорядочить B в порядке неубывания. Из каждой строки A вычесть B. Печатать A, B.

14. Заданы массивы A(4,4), B(4). Вычислить C=AB. Упорядочить C в порядке невозрастания. Из столбцов A вычесть C. Печатать A, B, C.

15. Задан массив B(4). Вычислить C(4) по формулам:

$$c_1 = \max(b_i), \quad c_2 = \min(b_i), \quad c_3 = c_1 + c_2, \quad c_4 = c_1 - c_2.$$

Вычислить элементы D(4,4) по формулам:

$$d_{ij} = \begin{cases} b_i - c_j, & b_i > 0 \\ b_i + c_j, & b_i \leq 0 \end{cases}.$$

Печатать B, C, D.

16. Заданы массивы A(4,4), B(4). Упорядочить строки A по неубыванию значений первых элементов строк. Вычислить элементы вектора C(4) по формуле  $c_i = a_{ii} - b_i$ . Печатать A, B, C.

17. Задан массив B(4). Вычислить элементы матрицы D(4,4) по формулам:

$$d_{ij} = \begin{cases} \sin(b_i + b_j), & 0 \leq |b_i + b_j| < \pi \\ \cos(b_i + b_j), & \pi \leq |b_i + b_j| < 2\pi \\ \sin(b_i - b_j), & 2\pi \leq |b_i + b_j| \end{cases}.$$

Найти максимальные элементы D и B, поменять их местами. Печатать B, D.

18. Задан массив B(4). Вычислить элементы матрицы D(4,4) по формулам. Из всех элементов D вычесть максимальный элемент B. Вычислить C=DB. Печатать B, C, D.

$$d_{ij} = \begin{cases} 2(b_i + b_j)^2, & b_i + b_j < -2 \\ 4(b_i + b_j), & -2 \leq b_i + b_j \leq 2 \\ \sqrt{b_i + b_j}, & b_i + b_j > 2 \end{cases}.$$

19. Задан массив A(4,4). Вычислить элементы вектора C(4) по формулам:

$$c_1 = \max(a_{ij}), \quad c_2 = (\max(a_{ij}) + \min(a_{ij})) / 2,$$

$$c_3 = (\max(a_{ij}) - \min(a_{ij})) / 2, \quad c_4 = \min(a_{ij}).$$

Вычислить  $P=AC$ . Из диагональных элементов  $A$  вычесть вектор  $P$ . Печатать  $A, C, P$ .

20. Заданы массивы  $A(4,4), B(4)$ . В матрице  $A$  найти минимальный и максимальный элементы, поменять их местами. Вычислить  $C=AB$ . Печатать  $A, B, C$ .

21. Заданы массивы  $A(4,4), B(4)$ . В массиве  $B$  найти максимальный и минимальный элементы, поменять их местами. Вычислить  $C=AB$ . Из диагональных элементов  $A$  вычесть вектор  $C$ . Печатать  $A, B, C$ .

22. Задан массив  $A(4,4)$ . Из диагональных элементов  $A$  вычесть максимальный элемент  $A$ . Вычислить вектор  $C(4)$  как сумму 1-й и 3-й строк  $A$ . Вычислить  $P=AC$ . Печатать  $A, C, P$ .

23. Заданы массивы  $A(4,4), B(4)$ . Вычислить  $C=AB$ . Упорядочить  $C$  в порядке неубывания. Из диагональных элементов  $A$  вычесть вектор  $C$ . Печатать  $A, B, C$ .

24. Заданы массивы  $A(4,4), B(4)$ . Вычислить элементы массива  $C(4)$  как сумму элементов в соответствующих строках  $A$ . Из диагональных элементов  $A$  вычесть  $C$ . Вычислить  $P=AC$ . Печатать  $A, C, P$ .

25. Задан массив  $A(4,4)$ . Вычислить элементы массива  $D(4,4)$  по формулам. Вычислить  $F=AD$ . Печатать  $A, D, F$ .

$$d_{ij} = \begin{cases} a_{ij}/2, & i < j \\ 2a_{ij}^2, & i \geq j \end{cases}.$$

26. Задан массив  $B(4)$ . Вычислить элементы массива  $C(4)$  по формуле  $c_k = \sum_{i=1}^k b_i$ . Вычислить элементы массива  $D(4,4)$  по формуле  $d_{ij} = b_i c_j$ . Печатать  $B, C, D$ .

27. Задан массив  $B(4)$ . Вычислить элементы массива  $C(4)$  по формуле  $c_k = \prod_{i=1}^k b_i$ . Вычислить элементы массива  $D(4,4)$  по формуле  $d_{ij} = (b_i + c_j)^2$ . Печатать  $B, C, D$ .

28. Заданы массивы  $A(4,4)$ ,  $B(4)$ . Вычислить  $C=AB$ . Из первой строки  $A$  вычесть  $B$ , из второй строки  $A$  вычесть вектор  $C$ . Печатать  $A$ ,  $B$ ,  $C$ .

29. Заданы массивы  $A(4,4)$ ,  $B(4)$ . Найти максимальный и минимальный элементы  $B$  и поменять их местами. Вычислить  $C=AB$ . Печатать  $A$ ,  $B$ ,  $C$ .

30. Задан массив  $B(4)$ . Вычислить элементы массива  $C(4)$  по формуле:  $c_i = i - \sin(0.4i + 1)$ . Найти и поменять местами максимальные элементы  $B$  и  $C$ . Вычислить элементы массива  $D(4,4)$  по формуле  $d_{ij} = \sqrt{|b_i + c_j|}$ . Печатать  $B$ ,  $C$ ,  $D$ .

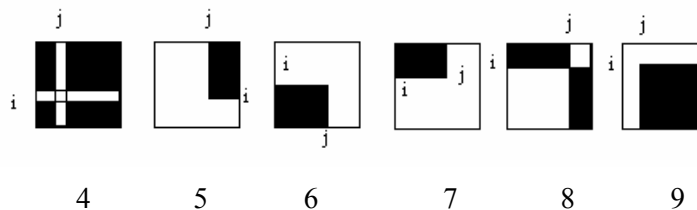
#### ЗАДАЧИ К ЗАЩИТЕ ЛАБОРАТОРНОЙ РАБОТЫ № 4

1. Даны  $A(n,n+1)$ ,  $B(n+1)$ ,  $C(n+1)$ . Получить  $D(n+1, n+2)$  вставкой в  $A$  после строки с номером  $p$  вектора  $B$  и последующей вставкой после столбца с номером  $g$  вектора  $C$ .

2. Дана  $A(n,m)$ . Получить  $D(n,m)$  перестановкой в  $A$  столбцов – первого с последним, второго с предпоследним и т.д.

3. Дана  $A(n,m)$ . Получить  $D(n,m)$  перестановкой в  $A$  строк – первой с последней, второй с предпоследней и т.д.

4-9. Дана  $A(n,m)$ . Получить  $D(n,m)$ . Каждый элемент  $d_{ij}$  равен сумме элементов матрицы  $A$  в заштрихованной области:



10. Дана  $A(n,m)$ . Получить  $D(n-1,m-1)$  путем выбрасывания из  $A$  строки и столбца, на пересечении которых расположен максимальный элемент  $A$ .

11. Дана действительная матрица  $A(n,m)$ . Найти сумму наибольших значений элементов ее строк.

12. Дано действительное число  $x$ . Получить  $A(n,n)$ :

$$\begin{bmatrix} 1 & x & x^2 & \dots & x^{n-1} \\ x^{n-1} & 1 & x & \dots & x^{n-2} \\ x^{n-2} & x^{n-1} & 1 & \dots & x^{n-3} \\ \dots & \dots & \dots & \dots & \dots \\ x & x^2 & x^3 & \dots & 1 \end{bmatrix}.$$

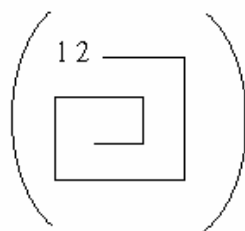
13. Дан вектор  $B(n)$ . Получить  $A(n,n)$ :

$$\begin{bmatrix} b_1 & b_2 & b_3 & \dots & b_n \\ b_2 & b_3 & b_4 & \dots & b_1 \\ \dots & \dots & \dots & \dots & \dots \\ b_n & b_1 & b_2 & \dots & b_{n-1} \end{bmatrix}.$$

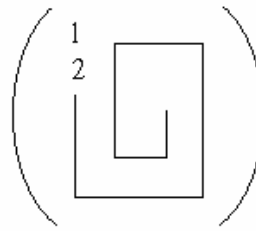
14. Дан вектор  $B(n)$ . Получить  $A(n,n)$ :

$$\begin{bmatrix} b_1 & b_2 & b_3 & \dots & b_n \\ 0 & b_1 & b_2 & \dots & b_{n-1} \\ \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & b_1 \end{bmatrix}.$$

15-16. Получить матрицу  $A(7,7)$ , элементами которой являются числа  $1,2,3,4,\dots,49$



15



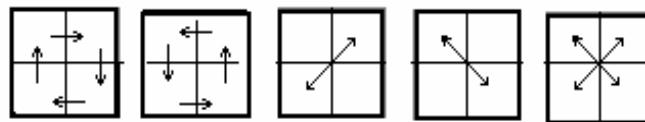
16

17-25. Дана матрица  $A(n,n)$ . Найти наибольший ее элемент в заштрихованной области



17      18      19      20      21      22      23      24      25

26-30. Дана матрица  $A(2n,2n)$ . Получить  $D(2n,2n)$ , переставляя блоки  $n \times n$  матрицы  $A$ :



26      27      28      29      30

### ЛИТЕРАТУРА

1. *Бартеньев О.В.* Современный FORTRAN. – М.: Диалог-МИФИ, 2000. – 447 с.
2. *Бартеньев О.В.* FORTRAN для студентов. – М.: Диалог-МИФИ, 1999. – 400 с.
3. *Рыжиков Ю.И.* Программирование на Фортране PowerStation для инженеров. Практическое руководство. – С.-Петербург: Корона принт, 1999. – 159 с.
4. *Немнюгин М.А., Стесик О.Л.* Современный Фортран. Самоучитель. – СПб.: БХВ-Петербург, 2004. – 496 с.
5. *Бартеньев О.В.* Visual Fortran: новые возможности. – М.: Диалог-МИФИ, 1999. – 301 с.
6. *Штыков В.В.* FORTRAN & WIN32 API. Создание программного интерфейса для Windows средствами современного Фортрана. – М.: Диалог-МИФИ, 2001. – 302 с.
7. *Симонович С.В.* Информатика. Базовый курс. – СПб.: Питер, 2007. – 640 с.

## **ПРИЛОЖЕНИЕ**

Министерство образования и науки Российской Федерации  
НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

---

### Лабораторные работы по курсу **“Информатика”**

Факультет летательных аппаратов  
Группа: ПС-81  
Студент: Иванов И.И.  
Преподаватель: Петров П.П.

НОВОСИБИРСК  
2008



## СОДЕРЖАНИЕ

Общие требования к оформлению лабораторных работ .....	3
Лабораторная работа № 1. ТАБУЛИРОВАНИЕ ФУНКЦИЙ .....	3
Лабораторная работа № 2. ПРОГРАММИРОВАНИЕ ВЕТВЯЩИХСЯ ВЫЧИСЛИТЕЛЬНЫХ ПРОЦЕССОВ .....	17
Лабораторная работа № 3. ПРОГРАММИРОВАНИЕ ЦИКЛИЧЕСКИХ ВЫЧИСЛИТЕЛЬНЫХ ПРОЦЕССОВ .....	30
Лабораторная работа № 4. ИСПОЛЬЗОВАНИЕ МАССИВОВ .....	39
Литература .....	54
Приложение .....	55

## ИНФОРМАТИКА И ПРОГРАММИРОВАНИЕ

### Методические указания

Выпускающий редактор *И.П. Брованова*  
Корректор *Л.Н. Кинит*  
Компьютерная верстка *Л.А. Веселовская*

---

Подписано в печать 10.09.2008. Формат 60 × 84 1/16. Бумага офсетная. Тираж 250 экз.  
Уч.-изд. л. 3,25. Печ. л. 3,5. Изд. № 144. Заказ № . Цена договорная

---

Отпечатано в типографии  
Новосибирского государственного технического университета  
630092, г. Новосибирск, пр. К. Маркса, 20