

«

»

“ ”

“ ” . . . . .  
\_\_\_\_\_ .

# РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ Информатика

: 09.03.02

, :

: 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>		

( ): 09.03.02

219 12.03.2015 ., : 30.03.2015 .

: 1,

( ): 09.03.02

, 2/1 20.06.2017

, 6 21.06.2017

:

, . . .

:

, . . . . . . . .

:

. . .

# 1.

1.1

<b>Компетенция ФГОС: ОК.10</b> способность к письменной, устной и электронной коммуникации на государственном языке и необходимое знание иностранного языка; <i>в части следующих результатов обучения:</i>	
3.	,
3.	- ,
<b>Компетенция ФГОС: ОПК.4</b> пониманием сущности и значения информации в развитии современного информационного общества, соблюдение основных требований к информационной безопасности, в том числе защите государственной тайны; <i>в части следующих результатов обучения:</i>	
2.	
<b>Компетенция ФГОС: ПК.2</b> способность проводить техническое проектирование; <i>в части следующих результатов обучения:</i>	
2.	
<b>Компетенция ФГОС: ПК.34</b> монтажно-наладочная деятельность: способность к установке, отладке программных и настройке технических средств для ввода информационных систем в опытную и промышленную эксплуатацию; <b>ПК.35</b> способность проводить сборку информационной системы из готовых компонентов; <i>в части следующих результатов обучения:</i>	
1.	
<b>Компетенция ФГОС: ПК.5</b> способность проводить моделирование процессов и систем; <i>в части следующих результатов обучения:</i>	
2.	-

# 2.

2.1

( , , , )	
-----------	--

<b>.2. 2</b>	
1. уметь применять основные методы, способы и средства получения, хранения и переработки информации с помощью компьютеров и компьютерных средств	; ;
2. Основные архитектуры компьютеров	;
<b>.4. 2</b>	
3. уметь пользоваться наиболее распространенными офисными и математическими пакетами прикладных программ	; ;
4. Знать роль информации в развитии современного информационного общества	;
<b>.5. 2</b> -	
5. Знать инструменты и методы моделирования бизнес-процессов организации	
<b>.10. 3</b> ,	
6. знать правовые основы информационной безопасности и принципы защиты авторского права на программные продукты	; ;
7. знать сущность и значение информации в развитии современного общества, опасности и угрозы, возникающие в этом процессе	; ;
<b>.10. 3</b> - ,	

8.уметь использовать специализированные программные средства при решении профессиональных задач	;	;
9.уметь использовать языки и системы программирования для решения профессиональных задач	;	
10.уметь оценивать состояние и тенденции развития информационных технологий и информатики в современном обществе	;	;
11.уметь использовать элементарные навыки алгоритмизации и программирования на одном из языков высокого уровня как средство программного моделирования изучаемых объектов и процессов	;	;
12.уметь проводить библиографическую и информационно-поисковую работы, использовать ее результаты при решении профессиональных задач и оформлении научных трудов	;	
13.уметь выполнять патентный поиск	;	
14.уметь проводить настройку программного обеспечения	;	;
15.уметь выполнять инсталляцию программного обеспечения	;	;
<b>.34 .35. 1</b>		
16. Знать как происходит инсталляция программных средств		;
<b>.10. 3</b> - ,		
17.уметь осуществлять поиск информации в локальных и глобальных сетях	;	;
18.владеть персональным компьютером как средством управления информацией	;	;

### 3.

#### 3.1

		,	.		
<b>: 1</b>					
<b>:</b>					
1.	0	2	13, 14, 15, 18, 2, 3, 4, 5, 7		
2.	0	2	1, 11, 18, 7		
3.	0	2	10, 12, 15		
<b>:</b>					
4.	0	2	10, 12, 18, 6, 7		
5.	0	2	1, 18, 6, 7		
6.	0	2	17, 18, 3, 7, 8, 9		
7.	0	2	1, 10, 12, 18		
8.	0	2	13, 14, 18		
9. -	0	2	11, 18, 6, 8		

: 1				
:				
1.	0	4	1, 18, 3, 6, 7	
2.	0	4	1, 10, 14, 15, 16, 17, 18, 3, 6, 7	
3.	2	4	11, 18, 8	
4.	0	4	10, 11, 18	
5.	4	8	1, 10, 11, 18	
6.	4	8	1, 10, 11, 18	
7.	4	4	10, 11, 18, 6	
8.	4	6	1, 10, 11, 18	
9.	0	4	10, 11, 18	
10.	0	4	1, 10, 11, 18	
11.	0	4	1, 10, 11, 18	

## 4.

: 1				
1		1, 3, 7	35	4
<p>1 : 220200 230100) / . . . - ; [ : . . . .]. - , 2009. - 33, [2] : .. - : <a href="http://elibrary.nstu.ru/source?bib_id=vtls000116740">http://elibrary.nstu.ru/source?bib_id=vtls000116740</a></p>				
2		1, 10, 11, 12, 13, 14, 15, 16, 17, 18, 2, 3, 4, 6, 7, 8, 9	14	4
<p>1 : 220200 230100) / . . . - ; [ : . . . .]. - , 2009. - 33, [2] : .. - : <a href="http://elibrary.nstu.ru/source?bib_id=vtls000116740">http://elibrary.nstu.ru/source?bib_id=vtls000116740</a></p>				
3		1, 6	8	5

. . . 1 : 1  
 ( 220200 230100) / . . . - ;  
 [ . . . .]. - , 2009. - 33, [2] . : .. - :  
[http://elibrary.nstu.ru/source?bib\\_id=vtls000116740](http://elibrary.nstu.ru/source?bib_id=vtls000116740)

## 5.

, ( . 5.1).

5.1

	-
	e-mail
	e-mail
	;

5.2

1	
<b>Краткое описание применения:</b> Выполнение задания в группе	

## 6.

( ), - 15- ECTS.  
 . 6.1.

6.1

<b>: 1</b>	
<i>Подготовка к занятиям:</i>	
<i>Лекция:</i>	10
<i>Лабораторная:</i>	25
<i>РГЗ:</i>	25
<i>Экзамен:</i>	40

		/		
.10	3. ,	+	+	+
	3. - ,	+	+	+
.4	2.	+	+	+
.2	2.	+	+	+
.34 .35	1.	+		
.5	2. -			+

1

## 7.

1. Подбельский В. В. Программирование на языке Си : учебное пособие для вузов / В. В. Подбельский, С. С. Фомин. - М., 2005. - 600 с. : ил., табл.
2. Подбельский В. В. Язык Си++ : [учебное пособие для вузов по направлениям "Прикладная математика" и "Вычислительные машины, комплексы, системы и сети"] / В. В. Подбельский. - М., 2007. - 559 с. : ил., табл.
3. Романов Е. Л. Си/Си ++. От дилетанта до профессионала [Электронный ресурс] : электронное учебное пособие : для 1-2 курсов направления 230100 "Информатика и вычислительная техника / Романов Е. Л. - Новосибирск, 2010. - 1 электрон. опт. диск (CD-ROM). - Загл. с этикетки диска. - Рег. свидетельство №18891. - Режим доступа: [http://elibrary.nstu.ru/source?bib\\_id=vtls000134024](http://elibrary.nstu.ru/source?bib_id=vtls000134024)

1. Крупник А. Изучаем Си / А. Крупник. - СПб., 2001. - 251, [1] с. : ил.

2. Язык Си для профессионалов. - М., 1991. - 383 с.

1. ЭБС НГТУ : <http://elibrary.nstu.ru/>

2. ЭБС «Издательство Лань» : <https://e.lanbook.com/>

3. ЭБС IPRbooks : <http://www.iprbookshop.ru/>

4. ЭБС "Znanium.com" : <http://znanium.com/>

5. :

## 8.

### 8.1

1. Программирование. Ч. 1 : методические указания к лабораторным работам для 1 курса АВТФ (направления 220200 и 230100) дневной формы обучения / Новосиб. гос. техн. ун-т ; [сост.: Г. П. Голодных и др.]. - Новосибирск, 2009. - 33, [2] с. : табл. - Режим доступа: [http://elibrary.nstu.ru/source?bib\\_id=vtls000116740](http://elibrary.nstu.ru/source?bib_id=vtls000116740)
2. Родников В. В. Лабораторные работы по курсу "Информатика" [Электронный ресурс] : учебно-методическое пособие / В. В. Родников ; Новосиб. гос. техн. ун-т. - Новосибирск, [2012]. - Режим доступа: [http://ciu.nstu.ru/fulltext/unofficial/2012/lib\\_397\\_1329632332.doc](http://ciu.nstu.ru/fulltext/unofficial/2012/lib_397_1329632332.doc). - Загл. с экрана.
3. Информатика. Ч. 1 : методические указания к лабораторным работам для студентов 1 курса факультета РЭФ специальностей "Радиотехника" и "Бытовая радиоэлектронная аппаратура" дневного отделения / Новосиб. гос. техн. ун-т ; [сост. В. В. Родников, О. Н. Шлыкова]. - Новосибирск, 2004. - 43 [1] с. : ил. - Режим доступа: <http://www.library.nstu.ru/fulltext/metodics/2004/2661.rar>
4. Информатика. Ч. 2 : методические указания к лабораторным работам для 1 курса РЭФ специальностей "Радиотехника", "Бытовая радиоэлектронная аппаратура" и "Радиосвязь, радиовещание и телевидение" дневного отделения / Новосиб. гос. техн. ун-т ; [сост. В. В. Дуркин, О. Н. Шлыкова]. - Новосибирск, 2007. - 41, [2] с.. - Режим доступа: <http://www.library.nstu.ru/fulltext/metodics/2007/3423.rar>

### 8.2

- 1 Microsoft Windows
- 2 Visual Studio 2015
- 3 Microsoft Office

## 9.

-

1	( - , , )	

1	( Internet )	



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

Кафедра систем сбора и обработки данных

“УТВЕРЖДАЮ”  
ДЕКАН АВТФ  
к.т.н. Рева И. Л.

“    ”    \_\_\_\_\_ г.

## ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

УЧЕБНОЙ ДИСЦИПЛИНЫ

**Информатика**

Образовательная программа: 09.03.02 Информационные системы и технологии

Факультет автоматики и вычислительной техники

# Обобщенная структура фонда оценочных средств учебной дисциплины

Тема	Код формируемой компетенции	Знания/умения	Контролирующее мероприятие (экзамен, зачет, курсовой проект и т.п.)
Подготовка и выполнение программ	ОК.10	у8. уметь проводить библиографическую и информационно-поисковую работы, использовать ее результаты при решении профессиональных задач и оформлении научных трудов	Лабораторная РГЗ Экзамен
История развития языков программирования	ОК.10 ОПК.4 ПК.2	з2. Знать роль информации в развитии современного информационного общества з2. Основные архитектуры компьютеров з5. знать сущность и значение информации в развитии современного общества, опасности и угроз, возникающие в этом процессе	Лабораторная РГЗ Экзамен

## Критерии оценки

Для допуска к экзамену необходимо :

1. Выполнить РГЗ (оценку за лабораторные работы выставляет ведущий лектор)
2. Выполнить лабораторные работы (оценку за лабораторные работы выставляет преподаватель ведущий цикл лабораторных работ)

Ниже приведены:

1. Задание на РГЗ
2. Примерные задания на экзамен

Лабораторные работы приведены в отдельном приложении.

Общая оценка определяется с весами указанными в рабочей программе в разделе «БРС и текущий контроль»

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

Кафедра ССОД

**Расчетно-графическая работа**

по дисциплине «Информатика»

Группа:

Студент:

Преподаватель: Гужов В.И.

Новосибирск 2016 г.

**Задание на выполнение РГЗ:**

1. Описать структуру представления чисел в форматах char, int, float, double.
2. Привести примеры представления чисел в форматах char, int, float, double.
3. Написать программу на языке C, которая должна
  - a. В двоичном виде представить число в формате char, int, float, double. (Для этого использовать объединение union)
  - b. Для чисел типа float разбить на три целых слова: знак, характеристика, мантисса. Определить порядок.
  - c. Для чисел типа float перевести в десятичный вид и вывести на печать отдельно целую часть и отдельно дробную.
  - d. Дробную часть переводить их двоичной формы в десятичную по правилам перевода умножением на 10.

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

Кафедра ССОД

Форма билета на экзамен

Дисциплина «Информатика»

**БИЛЕТ №**

**1. Теоретический вопрос**

**2. Практическое задание**

3. Составитель \_\_\_\_\_ доцент А.Э. Каспер

4. (подпись)

5. Заведующий кафедрой

6. \_\_\_\_\_ к.т.н., доцент Е.В. Прохоренко

7. (подпись)

8. « \_\_\_\_ » \_\_\_\_\_ 20\_\_ г.

Билет состоит из двух частей:

1. Теоретический вопрос
2. Задача

Ниже приведены примерные задания.



### *Экзаменационные вопросы (Теоретические вопросы)*

1. Системы счисления (Позиционные и непозиционные). Алгоритмы перевода из одной системы в другую. (Целые и дробные числа). Обратный и дополнительный код.
2. Типы данных. Переменные. Байтовые, целые, числа с плавающей запятой. Структура этих типов.
3. Операции и выражения. Перечисляемые константы. Преобразования типов (явные, неявные). Приоритеты и порядок выполнения.
4. Операторы языка. Составной оператор. Управляющие конструкции языка.
5. Условный оператор (if ... else ...). Переключатель (switch).
6. Операторы цикла (while , do while, for )
7. Операторы передачи управления (break , continue , goto , return )
8. Функции в С. Формальные и фактические параметры. Механизм передачи параметров. Возвращаемые значения. Использование указателей в качестве аргументов функции. Предварительное описание функций.
9. Адреса и указатели. Операции получения адреса и косвенная адресация. Отождествление массивов и указателей. Адресная арифметика.
10. Указатели на массивы. Массивы указателей и многомерные массивы. Динамическое выделение памяти под массивы. Инициализация указателей
11. Общая структура программы. Время существования и видимость переменных. Классы памяти
12. Структуры, объединения, перечисления
13. Рекурсивные вызовы функций. Аргументы командной строки.
14. Внешние объекты программы (extern). Статические переменные и функции (static). Регистровые переменные (register).

15. Препроцессор языка Си. (*#define #undef #include #if, #ifdef, #ifndef, #else, #elif u #endif*)

### *Задачи (написать программу на языке C)*

16. Написать на языке "C" или "C++" функцию, удовлетворяющую следующим требованиям, вызвать ее в программе main( )

Прототип: int f( char \*a, char \* b );

Параметры: a - указатель исходной строки; b - указатель результирующей строки

**Назначение:** отсортировать символы в строке в алфавитном порядке. Алгоритм: поиск минимального в исходной строке, запись в выходную строку.

17. Написать на языке "C" или "C++" функцию, удовлетворяющую следующим требованиям, вызвать ее в программе main( )

Прототип: void f( char \*a );

Параметры: a - указатель исходной строки

**Назначение:** Переставить в строке символы попарно слева и справа (1..n, 2..n-1 и т.д.);

18. Написать на языке "C" или "C++" функцию, удовлетворяющую следующим требованиям, вызвать ее в программе main( )

Прототип: void f( char \*a );

Параметры: a - указатель исходной строки

**Назначение:** Отсортировать символы в строке. Алгоритм: последовательно сравниваются пары соседних элементов  $x_k$  и  $x_{k+1}$  ( $k=0, \dots, n-2$ ), если  $x_k > x_{k+1}$ , то они переставляются.

19. Написать на языке "C" или "C++" функцию, удовлетворяющую следующим требованиям, вызвать ее в программе main( )

Прототип: void f( char \*a );

Параметры: a - указатель исходной строки

**Назначение:** Вывести на экран в первой строке нечетные символы, во второй четные символы исходной строки.

20. Написать на языке "C" или "C++" функцию, удовлетворяющую следующим требованиям, вызвать ее в программе main( ).

Прототип: void f( char \*a );

Параметры: a - указатель исходной строки

**Назначение:** Вывести на экран сначала цифры, затем буквы, затем остальные символы исходной строки.

21. Написать на языке "C" или "C++" функцию, удовлетворяющую следующим требованиям, вызвать ее в программе main( ).

Прототип: void f( char \*a );

Параметры: a - указатель исходной строки

**Назначение:** Для каждого символа исходной строки найти число вхождений его в строку и напечатать в столбец (символ, частота вхождения).

22. Написать на языке "C" или "C++" функцию, удовлетворяющую следующим требованиям, вызвать ее в программе main( )

Прототип: void f( char \*a , char \* b);

Параметры: a - указатель исходной строки

b - указатель результирующей строки

**Назначение:** Исключить из исходной строки все пробелы и записать в результирующую строку

23. Написать на языке "C" или "C++" функцию, удовлетворяющую следующим требованиям, вызвать ее в программе main( )

Прототип: int f( int \*mat, int n, int m, int &min, int &max );

Параметры: mat - указатель исходной матрицы

n – количество строк

m – количество столбцов

min – минимальное значение

max – максимальное значение

**Назначение:** Вычисление максимального и минимального значений элементов матрицы

24. Написать на языке "C" или "C++" функцию, удовлетворяющую следующим требованиям, вызвать ее в программе main( )

Прототип: int f( int \*mat, int n, int m, int & A, int & B );

Параметры: mat - указатель исходной матрицы

n – количество строк

m – количество столбцов

A – число элементов > 0

B – число элементов < 0

**Назначение:** Вычисление числа элементов больше нуля и меньше нуля

25. Написать на языке "C" или "C++" функцию, удовлетворяющую следующим требованиям, вызвать ее в программе main( )

Прототип: int f( int \*mat, int n );

Параметры: mat - указатель квадратной матрицы

n – количество строк, столбцов

m – количество столбцов

**Назначение:** Заполнение матрицы последовательностью чисел от 0 до  $n^2$ , расположенными по спирали

1	2	3	4	5	6
20	21	22	23	24	7
19	32	33	34	25	8
18	31	36	35	26	9
17	30	29	28	27	10
16	15	14	13	12	11

26. Написать на языке "C" или "C++" функцию, удовлетворяющую следующим требованиям, вызвать ее в программе main( )

Прототип: void f( int \*A, int \*B, int n, int m );

Параметры: A - указатель исходной матрицы

B - указатель транспонированной матрицы

n – количество строк исходной матрицы

m – количество столбцов исходной матрицы

**Назначение:** Нахождение транспонированной матрицы.

Транспонирование заключается в обмене местами строк и столбцов матрицы. Элемент, расположенный в i-той строке и j-том столбце помещается в j-тую строку и i-тый столбец.

27. Написать на языке "C" или "C++" функцию, удовлетворяющую следующим требованиям, вызвать ее в программе main( )

Прототип: void f( int \*A, int \*X, int \*Y, int n, int m );

Параметры: A - указатель матрицы

X - указатель вектора (одномерного массива)

Y – указатель результирующего вектора

n – количество строк матрицы

m – количество столбцов матрицы

**Назначение:** вычисление произведения матрицы на вектор.

В результате умножения матрицы A[m][n] на вектор X[n] получается вектор Y[m], компоненты которого вычисляются по формуле:  $y[i] = \sum (a[i][j] * x[j])$ , (i = 0, 2, ..., m-1)

28. Написать на языке "C" или "C++" функцию, удовлетворяющую следующим требованиям, вызвать ее в программе main( )

Прототип: void f( int \*A, int \*B, int \*C, int n, int m, int l );

Параметры: A - указатель первой матрицы

B - указатель второй матрицы

C - результирующей матрицы

n – количество строк первой матрицы

m – количество столбцов первой матрицы, количество строк второй матрицы

l – количество столбцов второй матрицы

**Назначение:** вычисление произведения 2-х матриц (строка на столбец).

В результате умножения двух матриц A[m][n] и B[n][l] получается матрица C[m][l], элементы которой находятся по следующей формуле:  $C[i][j] = \sum a[i][j] * b[k][j]$

29. Написать на языке "C" или "C++" функцию, удовлетворяющую следующим требованиям, вызвать ее в программе main( )

Прототип: int f( int \*A, int m );

Параметры: A - указатель квадратной матрицы

m – количество строк, столбцов матрицы

**Назначение:** вычисление среднего арифметического элементов, лежащих ниже диагонали

---

30. Написать на языке "C" или "C++" функцию, удовлетворяющую следующим требованиям, вызвать ее в программе main( )

Прототип: `int f( int *a, int n );`

Параметры: `a` - указатель исходного массива, `n` - размер исходного массива.

**Назначение:** в исходном массиве все элементы, встречающиеся больше одного раза заменить на нули.

31. Написать на языке "C" или "C++" функцию, удовлетворяющую следующим требованиям, вызвать ее в программе `main( )`

Прототип: `int f( int *a, int n );`

Параметры: `a` - указатель исходного массива, `n` - размер исходного массива.

**Назначение:** Отсортировать элементы массива по убыванию методом пузырька

Алгоритм : последовательно сравниваются пары соседних элементов  $x_k$  и  $x_{k+1}$  ( $k=0, n-2$ ), если  $x_k < x_{k+1}$  , то они переставляются.

32. Написать на языке "C" или "C++" функцию, удовлетворяющую следующим требованиям, вызвать ее в программе `main( )`

Прототип: `int f( int *a, int n );`

Параметры: `a` - указатель исходного массива, `n` - размер исходного массива.

**Назначение:** Отсортировать элементы массива по следующему алгоритму: отыскивается максимальный элемент и переносится в конец массива, затем метод переносится ко всем элементам, кроме последнего и т.д.

---

33. Написать на языке "C" или "C++" функцию, удовлетворяющую следующим требованиям, вызвать ее в программе `main( )`

Прототип: `int f( int M, int N );`

Параметры: `M, N` - целые числа

**Назначение:** Нахождение наибольшего общего делителя двух чисел `N` и `M` по рекуррентному соотношению

$N_k = N_{k-2} - \text{INT}(N_{k-2} / N_{k-1}) N_{k-1} \quad k=2,3 \dots$

$N_0 = \max(|N|, |M|) \quad N_1 = \min(|N|, |M|)$

Если  $N_k = 0 \Rightarrow \text{НОД} = N_{k-1}$

( $N=23345 \quad M=9135 \Rightarrow 1015 \quad N=238 \quad M=347 \Rightarrow 34$ )

34. Написать на языке "C" или "C++" функцию, удовлетворяющую следующим требованиям, вызвать ее в программе `main( )`

2.1. Прототип: `int f( int *a, int n );`

2.2. Параметры: `a` - указатель исходного массива, `n` - размер исходного массива.

2.3. Назначение: в исходном массиве удаляются все элементы, значения которых меньше, чем количество их повторов в массиве. Порядок следования остальных элементов массива не изменяется.

2.4. Возвращаемое значение: количество оставшихся элементов.

2.5. Пример использования:

```
int a[] = { 1, 2, 3, 4, 2, 4, 6, 2, 4, 7, 1, 1, 5 };
```

```
int n;
```

```
printf( "n = %d\n", n = f( a, sizeof(a)/sizeof(int)) );
```

```
for( int i = 0 ; i < n ; printf( "%d ", a[i++] ) );
```

```
// вывод на экран:
```

```
//
```

```
// n = 7
// 3 4 4 6 4 7 5
```

---

35. Написать на языке "C" или "C++" функцию, удовлетворяющую следующим требованиям, вызвать ее в программе main( )

2.1. Прототип: int f( int\* a, int n, int k );  
2.2. Параметры: a - указатель исходного массива, n - размер массива, k - размер сдвига.  
2.3. Назначение: исходный массив преобразуется в результате циклического сдвига на указанное число позиций. "Циклически" означает, что при выходе за верхнюю границу массива запись производится в позиции 0, 1 и т.д., а при выходе за нижнюю - в позиции n-1, n-2 и т.д.  
2.4. Возвращаемое значение: свой первый параметр.  
2.5. Пример использования:  
int i, x[] = { 3, 6, 4, 1, 8, 2, 5, 7, 0 };  
f( x, sizeof(x)/sizeof(int), -3 );  
for( i = 0 ; i < sizeof(x)/sizeof(int) ; ++i ) printf( "%d ", x[i] );  
f( x, sizeof(x)/sizeof(int), 2 ); printf( "\n" );  
for( i = 0 ; i < sizeof(x)/sizeof(int) ; ++i ) printf( "%d ", x[i] );  
// вывод на экран:  
//  
// 1 8 2 5 7 0 3 6 4  
// 6 4 1 8 2 5 7 0 3  
//

36. Написать на языке "C" или "C++" функцию, удовлетворяющую следующим требованиям, вызвать ее в программе main( )

2.1. Прототип: int f( char \*a, float \*b );  
2.2. Параметры: a - указатель исходной строки, b - указатель буфера.  
2.3. Назначение: в исходной строке ищутся и распечатываются в столбик все последовательности литер, представляющие записи целых чисел или действительных чисел в формате <целая часть>.<дробная часть>. Сами числа в формате float последовательно записываются в буфер.  
2.4. Возвращаемое значение: количество найденных и записанных чисел.  
2.5. Пример использования:  
char s[] = "12.45, 678.99, -34 0.0004 ";  
float B[100];  
printf( "\nЗаписано чисел: %d", f( s, B ));  
// вывод на экран:  
//  
// 12.450000  
// 678.99000  
// -34.000000  
// 0.000400  
// Записано чисел: 4  
//

---

37. Написать на языке "C" или "C++" функцию, удовлетворяющую следующим требованиям, вызвать ее в программе main( )

2.1. Прототип: int\* f( int\* a, int n );  
2.2. Параметры: a - указатель исходного массива, n - размер массива.  
2.3. Назначение: исходный массив преобразуется по следующему алгоритму: все нечетные элементы массива группируются в порядке возрастания в начале массива, а четные в порядке убывания в конце.  
2.4. Возвращаемое значение: указатель первого четного элемента в преобразованном массиве.  
2.5. Пример использования:  
int x[] = { 3, 6, 4, 8, 2, 5, 7, 0 };  
int \*p = f( x, sizeof(x)/sizeof(int));  
printf( "n = %d\n", p-x );  
for( int i = 0 ; i < sizeof(x)/sizeof(int) ; ++i ) printf( "%d ", x[i] );  
// вывод на экран:  
//  
// n = 3  
// 3 5 7 8 6 4 2 0  
//

38. Написать на языке "C" или "C++" функцию, удовлетворяющую следующим требованиям, вызвать ее в программе main( )

2.1. Прототип: char\* f( char \*v1, char \*v2, char \*r, int n );  
2.2. Параметры: v1, v2 - указатели исходных строк, r - указатель результирующей строки, n - размер результирующей строки.  
2.3. Назначение: исходные строки являются символьным представлением натуральных чисел произвольной длины. Необходимо получить в результирующей строке такое же представление **суммы** этих чисел. Не уместающиеся в результирующей строке цифры пропадают. Остающиеся свободные позиции заполняются кодом нуля.  
2.4. Возвращаемое значение: указатель литеры, являющейся представлением первой значащей цифры результата.  
2.5. Пример использования:  
char a[] = "45789", b[] = "1111111", c[20];  
printf( "\n%s\n%s", f( a, b, c, sizeof c ), c );  
//  
// вывод на экран:  
//  
// 1156900  
// 000000000000001156900  
//

39. Написать на языке "C" или "C++" функцию, удовлетворяющую следующим требованиям, вызвать ее в программе main( )

Прототип: double f(int n );



Назначение:  $y = 1! + 2! + 3! + 4! + \dots + n!$

40. Написать на языке "C" или "C++" функцию, удовлетворяющую следующим требованиям, вызвать ее в программе main( )

Прототип: double f(double x, double e);

Найти функцию с точностью  $\epsilon$ , т.е. пока  $|f_{n+1} - f_n| < \epsilon$ . Определить абсолютную разность между табличной функцией и суммой ряда. Определить с какой точностью в машине определяется эта функция.

$$e^x = 1 + x + x^2/2! + x^3/3! + \dots$$

41. Написать на языке "C" или "C++" функцию, удовлетворяющую следующим требованиям, вызвать ее в программе main( )

Прототип: double f(double x, double e);

Найти функцию с точностью  $\epsilon$ , т.е. пока  $|f_{n+1} - f_n| < \epsilon$ . Определить абсолютную разность между табличной функцией и суммой ряда. Определить с какой точностью в машине определяется эта функция.

$$\sin(x) = x - x^3/3! + x^5/5! + \dots$$

42. Написать на языке "C" или "C++" функцию, удовлетворяющую следующим требованиям, вызвать ее в программе main( )

Прототип: double f(double x, double e);

Найти функцию с точностью  $\epsilon$ , т.е. пока  $|f_{n+1} - f_n| < \epsilon$ . Определить абсолютную разность между табличной функцией и суммой ряда. Определить с какой точностью в машине определяется эта функция.

$$\cos(x) = 1 - x^2/2! + x^4/4! - \dots$$

43. Написать на языке "C" или "C++" функцию, удовлетворяющую следующим требованиям, вызвать ее в программе main( )

Прототип: double f(double x, double e);

Найти функцию с точностью  $\epsilon$ , т.е. пока  $|f_{n+1} - f_n| < \epsilon$ . Определить абсолютную разность между табличной функцией и суммой ряда. Определить с какой точностью в машине определяется эта функция.

$$\ln(1+x) = x - x^2/2 + x^3/3 - \dots \quad |x| < 1$$

44. Написать на языке "C" или "C++" функцию, удовлетворяющую следующим требованиям, вызвать ее в программе main( )

Прототип: double f(double x, double e);

Найти функцию с точностью  $\epsilon$ , т.е. пока  $|f_{n+1} - f_n| < \epsilon$ . Определить абсолютную разность между табличной функцией и суммой ряда. Определить с какой точностью в машине определяется эта функция.

$$\cos(x) = 1 - x^2/2! + x^4/4! \dots$$

45. Написать на языке "C" или "C++" функцию, удовлетворяющую следующим требованиям, вызвать ее в программе main( )

Прототип: double f(double x, double e);

Найти функцию с точностью  $\epsilon$ , т.е. пока  $|f_{n+1}-f_n| < \epsilon$ . Определить абсолютную разность между табличной функцией и суммой ряда. Определить с какой точностью в машине определяется эта функция.

$$\operatorname{tg}(x) = \frac{x}{1 - \frac{x^2}{3 - \frac{x^2}{5 - \frac{x^2}{7 - \dots}}}}$$

46. Написать на языке "C" или "C++" функцию, удовлетворяющую следующим требованиям, вызвать ее в программе main( )

Прототип: double f(double e);

Найти функцию с точностью  $\epsilon$ , т.е. пока  $|f_{n+1}-f_n| < \epsilon$ . Определить абсолютную разность между табличной функцией и суммой ряда. Определить с какой точностью в машине определяется эта функция.

$$\operatorname{sqrt}(2) = 1 + \frac{1}{2 + \frac{1}{2 + \frac{1}{2 + \frac{1}{2 + \dots}}}}$$

47. Написать на языке "C" или "C++" функцию, удовлетворяющую следующим требованиям, вызвать ее в программе main( )

Прототип: double f(double e);

Найти функцию с точностью  $\epsilon$ , т.е. пока  $|f_{n+1}-f_n| < \epsilon$ . Определить абсолютную разность между табличной функцией и суммой ряда. Определить с какой точностью в машине определяется эта функция.

$$e = 2 + \frac{1}{2 + \frac{1}{3 + \frac{1}{4 + \frac{1}{5 + \dots}}}}$$

48. Написать на языке "C" или "C++" функцию, удовлетворяющую следующим требованиям, вызвать ее в программе main( )

Прототип: double f(double x, double e);

Найти функцию с точностью  $e$ , т.е. пока  $|f_{n+1} - f_n| < e$ . Определить абсолютную разность между табличной функцией и суммой ряда. Определить с какой точностью в машине определяется эта функция.

49. Написать на языке "C" или "C++" функцию, удовлетворяющую следующим требованиям, вызвать ее в программе main( )

Прототип: void f(int n);

Печать римских цифр

- |    |       |        |
|----|-------|--------|
| 1. | 1-I   | 10-X   |
| 2. | 2-II  | 50-L   |
| 3. | 3-III | 100-C  |
| 4. | 4-IV  | 500-D  |
| 5. | 5-V   | 1000-M |
| 6. | 6-VI  |        |

Соблюдая следующие правила:

7. Если младшая цифра стоит перед старшей она вычитается IV-4
8. Если младшая цифра стоит после старшей - складывается VI -5
9. Число представляется минимальным количеством знаков III-3, 4-IV, а не IIII.

50. Написать на языке "C" или "C++" функцию, удовлетворяющую следующим требованиям, вызвать ее в программе main( )

Прототип: int f(int n, int k);

Напечатать подряд все натуральные числа до n - 012345678910111213 Найти k-ую цифру последовательности

51. Написать на языке "C" или "C++" функцию, удовлетворяющую следующим требованиям

Найти все четырехзначные числа Армстронга. Натуральное число из n знаков является числом Армстронга, если сумма его цифр, возведенная в n-ю степень равна самому числу.  $153=1^3+5^3+3^3$

52. Написать на языке "C" или "C++" функцию, удовлетворяющую следующим требованиям

Вставить между цифрами 1,2,3,4...9 знак одной из операций +, -, \*, /, так, чтобы результат 8-операций равнялся 100. Имеется 8 решений. Распечатать их со знаками.

$$1*2+3*4*5-6+7+8-9=10$$

53. Написать на языке "C" или "C++" функцию, удовлетворяющую следующим требованиям

Совершенными числами называются числа, равные сумме своих делителей ( $6=1+2+3$ ). Найти все совершенные числа от 0 до 10000 и распечатать их вместе со своими делителями.

54. Написать на языке "C" или "C++" функцию, удовлетворяющую следующим требованиям, вызвать ее в программе main( )

Прототип: void f1(int n, \*char a); // a – массив, состоящий из 0 или 1. В main( ) массив распечатать

Перевести из десятичной системы в двоичную целое число

327	:	2	=	163	остаток	1
163	:	2	=	81		1
81	:	2	=	40		1
40	:	2	=	20		0
20	:	2	=	10		0
10	:	2	=	5		0
5	:	2	=	2		1
2	:	2	=	1		0
1	:	2	=	0		1

$$327_{10} = 101000111_2$$

55. Написать на языке "C" или "C++" функцию, удовлетворяющую следующим требованиям, вызвать ее в программе main( )

Прототип: void f1(double b, \*char a); // a – массив, состоящий из 0 или 1. В main( ) массив распечатать

Перевести из десятичной системы в двоичную дробное

$$0.125 \times 2 = 0.250 = 0 + 0.250$$

$$0.250 \times 2 = 0.5 = 0 + 0.5$$

$$0.500 \times 2 = 1.000 = 1 + 0.00$$

$$\mathbf{0.125_{10} = 0.001_2}$$

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

Кафедра систем сбора и обработки данных

“УТВЕРЖДАЮ”  
ДЕКАН АВТФ  
к.т.н., доцент И.Л. Рева  
“ ” Г.

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

**УЧЕБНОЙ ДИСЦИПЛИНЫ**

# Информатика

Образовательная программа: 09.03.02 Информационные системы и технологии, профиль: Информационные системы в промышленности и бизнесе

# 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(Р) и др.)	Промежуточная аттестация (экзамен, зачет)
ОК.10 способность к письменной, устной и электронной коммуникации на государственном языке и необходимое знание иностранного языка	з5. знать сущность и значение информации в развитии современного общества, опасности и угроз, возникающие в этом процессе	Изучение операционной системы История развития языков программирования Подготовка и выполнение программ	Отчет по лабораторной работе	Экзамен, вопросы 1-10
ОК.10	у8. уметь проводить библиографическую и информационно-поисковую работы, использовать ее результаты при решении профессиональных задач и оформлении научных трудов	Ввод-вывод Ввод-вывод файлов Изучение структуры встроенных типов (целые, числа с плавающей запятой) История развития языков программирования Массивы и структуры Нахождение суммы, произведения, факториалов Оформление программ Подготовка и выполнение программ Работа с динамическими массивами Работа с символами Работа со статическими массивами Решение квадратного уравнения Решение системы уравнений Системы счисления Установка системы программирования	Отчет по лабораторной работе	Экзамен, вопросы 11-15
ОПК.4 пониманием сущности и значения информации в развитии современного информационного общества, соблюдение основных требований к информационной безопасности, в том числе защите государственной тайны	з2. Знать роль информации в развитии современного информационного общества	История развития языков программирования Установка системы программирования	РГЗ	Экзамен, вопросы 10-15
ПК.2 способность проводить техническое проектирование	з2. Основные архитектуры компьютеров	Изучение операционной системы История развития языков программирования Оформление программ Установка системы программирования		Экзамен, вопрос 14

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

Промежуточная аттестация по дисциплине проводится в 1 семестре - в форме экзамена, который направлен на оценку сформированности компетенций ОК.10, ОПК.4, ПК.2.

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(Р)). Требования к выполнению РГЗ(Р), состав и правила оценки сформулированы в паспорте РГЗ(Р).

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.10, ОПК.4, ПК.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.



*Лабораторная работа 1*

**Ввод-вывод. Арифметические выражения. Циклы**

**1.1) Ввести с терминала x. Вычислить выражение. Распечатать результат..**

а)

$$y = \begin{matrix} 3.5 + & 7x - & 5.8x^2 & x < 5 \\ -17.3 + & 85x - & 13.8x^2 & x = 5 \\ -18.0 + & 17.85x + & x^2 & x > 5 \end{matrix}$$

б)

$$y = \begin{matrix} 3i & i = 0, 2, 4, 6, 8 \dots \\ 0.5i & i = 1, 3, 5, 7, 9 \dots \end{matrix}$$

**1.2) Нахождение сумм, произведений  $i=1,\dots,N$   $j=1,\dots,N$**

1.  $y = S i$        $y = \Pi i$
2.  $y = S (1/i)$        $y = \Pi (1/i)$
3.  $y = S (i/(i+1))$        $y = \Pi (i/(i+1))$
4.  $y = S S i j$        $y = \Pi \Pi 1/(i+j)$
5.  $y = S \Pi i j$        $y = \Pi S i j$

**1.3) Дана последовательность целых чисел 0,1,2,... Найти сумму чисел расположенных между min и max числами.**

**1.4) Напечатать таблицу**

а)

i / j	1	2	...	i
1	...	...	...	...
2	...	S i j	...	...

...	...	...	...	...
j	...	...	...	...

б)

i/j	1	2	...	i
1	...	...	...	...
2	...	$\pi_{ij}$	...	...
...	...	...	...	...
j	...	...	...	...

в)

x	sin (x)	cos (x)	tan (x)
0.000	0.000	1.000	0.000
0.100	...	...	...
...	...	...	...
1.000	...	...	...

### 1.5) Найти параметры треугольника

а) Найти стороны по координатам  $(x_1, y_1), (x_2, y_2), (x_3, y_3)$ ,

$$a = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}; b = \dots; c = \dots;$$

б) Найти площадь по формуле Герона  $s = \sqrt{p(p-a)(p-b)(p-c)}$

с) Определить высоты  $h = (2/a)\sqrt{p(p-a)(p-b)(p-c)}$

д) Заштриховать треугольник произвольной формы

е) Нарисовать треугольник и его высоты

**1.6) Решить квадратное уравнение  $ax^2 + bx + c = 0$**

$$x_{1,2} = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

Проверить при  $a=1.000$   $b=-200.0$   $c=1.000 \Rightarrow x_1=200. x_2=0.005$

При большом расхождении коэффициентов использовать правило Виста  $x_1 \cdot x_2 = c/a$ . Находим наибольший корень.

**1.7) Решить систему**

а) из 2 уравнений

$$a_1 x + b_1 y = c_1$$

$$a_2 x + b_2 y = c_2$$

б) из 3 уравнений

$$a_{1,1}x + a_{1,2}y + a_{1,3}z = b_1$$

$$a_{2,1}x + a_{2,2}y + a_{2,3}z = b_2$$

$$a_{3,1}x + a_{3,2}y + a_{3,3}z = b_3$$

с) из N уравнений методом Гауса

**1.8)**

1. Идет к-я секунда суток определить сколько прошло часов, минут, остаток секунд.
2. Определить полное количество часов, минут, секунд, прошедших от начала суток до того момента, когда часовая стрелка повернулась на  $f^\circ$  - градусов.
3. Определить угол между положением часовой стрелки от начала суток до ее положения в  $p$  часов,  $m$  минут,  $s$  секунд.
4. Нарисовать идущие часы со стрелками (время от таймера компьютера)

### **1.9) Нахождение факториала**

1.  $y = x!$
2.  $y = x! / N!$
3.  $y = (2n-1)!! = 1 \cdot 3 \cdot 5 \cdot (2n-1) \quad n > 0$
4.  $y = (2n)!! = 2 \cdot 4 \cdot (2n) \quad n > 0$
5.  $y = n!!$
6.  $y = 1! + 2! + 3! + 4! + \dots + n!$

### **1.10) Найти первый отрицательный член последовательности $\cos(\operatorname{ctg} n)$ $n=1,2,3\dots$**

### **1.11) Вычислить $k$ -й член последовательности при $x_0 = 1$**

$$x_n = n x_{n-1} + 1/n$$

### **1.12) Вычислить**

1.  $y = \sin 1 + \sin 1.1 + \sin 1.2 + \dots + \sin 2.$
2.  $y = \sqrt{3 + \sqrt{6 + \dots \sqrt{96 + \sqrt{99}}}}$
3.  $y = \cos(1 + \cos(2 + \dots + \cos(39 + \cos(40))))$
4. Вычислить  $p = (1 - 1/2^2) (1 - 1/3^2) (1 - 1/n^2) \quad n > 2$
5. Вычислить с точностью до  $\epsilon$  (Разность между найденным и предыдущим значениями меньше  $\epsilon$   $q = \sqrt{1 + \sqrt{1 + \sqrt{1 + \dots}}}$  )

### **1.13) Найти кол-во целых точек, расположенных внутри окружности радиуса $R$ .**

## **Лабораторная работа 2. Массивы**

### **2.1) Ввести массив $X(N)$ . Найти**

- а) Мат.ожидание  $M = (1/N) \sum X(i)$
- б) Дисперсию  $D = (1/(N-1)) \sum (X(i) - M)^2$

- c) Среднее квадратическое отклонение  $\sqrt{D}$

2.2) Ввести массив  $X(N)$ . Найти максимум и минимум.

2.3) Ввести массив  $X(N)$ . Переписать в массив  $B(N)$  в порядке возрастания, в  $C(N)$  в порядке убывания и выдать их на печать.

2.4) Ввести матрицу  $M \times N$  и выдать ее на печать и транспонированную матрицу.

2.5) Ввести матрицу и выдать на печать

- а) сумму и разность
- б) произведение (строка на столбец)
- c) произведение вектора на матрицу

2.6) Ввести матрицу и выдать на печать

- а) число элементов  $>$ ,  $<$ ,  $=$  нулю. Определить максимум и минимум.
- б)  $S$  элементов  $<0$ ,  $>0$ ,  $S$  элементов всей матрицы
- в)  $S$  элементов диагонали,  $S$  элементов по строкам,  $S$  элементов по столбцам
- г) Определить номер столбца на пересечении которых находится минимальный, максимальный элементы.

2.7) Найти среднее арифметическое элементов, лежащих ниже диагонали.

2.8) Получить вектор  $B[N]$  - каждый элемент которого среднее арифметическое элементов соответствующей строки и вектор  $C[M]$  - каждый элемент которого среднее арифметическое элементов соответствующего столбца.

2.9) Получить вектор  $B[N]$  - каждый элемент которого максимальный элемент соответствующей строки и вектор  $C[M]$  - каждый элемент которого максимальный элемент соответствующего столбца.

2.10) В матрице найти расстояние между максимальным и минимальным элементами  $R = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$

2.11) Заполнить квадратную таблицу ( $n \times n$ ) последовательными числами от 0 до  $n*n$ , расположенными по спирали.

1	2	3	4	5	6
20	21	22	23	24	7
19	32	33	34	25	8

18 31 36 35 26 9  
17 30 29 28 27 10  
16 15 14 13 12 11

2.12). Тоже, что и предыдущем случае, но рисовать каждую цифру по мере вычисления. (Используя функцию gotoxy()).

2.13) Упорядочить массив по возрастанию

- а) сортировка выбором: отыскивается максимальный элемент и переносится в конец массива, затем метод переносится ко всем элементам кроме последнего и т.д.
- б) сортировка обменом (метод пузырька): последовательно сравниваются пары соседних элементов  $x_k$  и  $x_{k+1}$  ( $k=1, n-1$ ), если  $x_k > x_{k+1}$ , то они переставляются.
- в) сортировка вставками: пусть первые  $k$  элементов уже упорядочены, берется  $(k+1)$  элемент и упорядочивается между  $k$  - элементами, чтобы упорядоченными были уже  $k+1$ .

2.14) Ввести массив, в каждой строке расположить элементы по возрастанию.

### *Лабораторная работа 3,4* *Ряды, цепные дроби*

3.1) Найти функцию с точностью  $\epsilon$ , т.е. пока  $|f_{n+1} - f_n| < \epsilon$ . Определить необходимое число членов ряда. Определить абсолютную разность между табличной функцией и суммой ряда. Определить с какой точностью в машине определяется эта функция.

1)  $e^x = 1 + x + x^2/2! + x^3/3! + \dots$

2)  $\sin(x) = x - x^3/3! + x^5/5! + \dots$

3)  $\cos(x) = 1 - x^2/2! + x^4/4! - \dots$

4)  $\ln(1+x) = x - x^2/2 + x^3/3 - \dots$   $|x| < 1$

5)  $y = \arctg(x) = x - x^3/3 + x^5/5 - \dots$   $|x| < 1$

6)  $\operatorname{ch} x = 1 + x^2/2! + x^4/4! + x^{2n}/(2n)! + \dots$

7)  $\operatorname{sh} x = x + x^3/3! + x^5/5! + x^{2n+1}/(2n+1)! + \dots$

3.2) Вычислить сумму ряда  $1 - 1/2 + 1/3 - \dots + 1/999 - 1/1000$

1. слева - направо
2. слева - направо, сначала все положительные, затем все отрицательные и вычесть из положительных отрицательные

3. справа - налево

4. справа - налево, сначала все положительные, затем все отрицательные и вычесть из положительных отрицательные

(0.603097183059945296917232371458)

3.3) Определить корень третьей степени от  $Z$  по итерационной формуле, пока  $|W_{n+1} - W_n| < \epsilon$

$$W_{n+1} = W_n + (1/3)(Z/W_n^2 - W_n)$$

начальное значение

$$Z/3, \text{ если } Z \geq 0$$

$$W_0 =$$

$$\sqrt[3]{Z} \text{ если } Z < 0$$

Использовать функцию  $F(w) = w + (1/3)(Z/w^2 - w)$

3.4) Вычислить квадратный корень из произвольного вещественного числа  $A > 0$  по итерационной формуле, пока  $|Y_{n+1} - Y_n| < \epsilon$

$$Y_{n+1} = 0.5(Y_n + A/Y_n) \text{ и полагая } Y_0 = A$$

3.5) Получение  $N$ -го числа Фибоначи, т.е. числа из последовательности

0 1 1 2 3 5 8 13 21 34

$$c_i = c_{i-1} + c_{i-2} \quad (c_1 = 0 \quad c_2 = 1) \quad n=10 \quad c=34$$

3.6) Дано  $k$  от 2 до 20. Найти коэффициенты  $k$ -го многочлена Чебышева по формуле

$$T_0(x)=1, T_1(x)=x \quad T_n(x)=2xT_{n-1}(x)-T_{n-2}(x)$$

3.7) Нахождение наибольшего общего делителя двух чисел  $N$  и  $M$  по рекуррентному соотношению

$$(N=23345 \quad M=9135 \Rightarrow 1015 \quad N=238 \quad M=347 \Rightarrow 34)$$

$$N_k = N_{k-2} - \text{INT}(N_{k-2} / N_{k-1}) N_{k-1} \quad k=2,3 \dots$$

$$N_0 = \max(|N|, |M|) \quad N_1 = \min(|N|, |M|)$$

$$\text{Если } N_k = 0 \Rightarrow \text{НОД} = N_{k-1}$$

3.8) Вычислить с точностью до  $\epsilon$  (Разность между найденным и предыдущим значениями меньше  $\epsilon$ )

$$y = 1 + \frac{1}{x + \frac{1}{x + \frac{1}{x + \frac{1}{x + \frac{1}{x}}}}} \\ tg(x) = \frac{x^7 - x^5 + x^3 - x}{x^8 + 1} \\ y = 1 + \frac{1}{1 + \frac{1}{1 + \frac{1}{1 + \frac{1}{1 + \frac{1}{1}}}}} \\ sqrt(2)=1 + \frac{1}{2 + \frac{1}{2 + \frac{1}{2 + \dots}}} \\ e=2+\frac{2}{2+\frac{3}{2+\frac{4}{\ddots}}}$$



$$\begin{array}{lcl}
 42. & & 3 + \frac{\quad}{\quad} \\
 43. & & \quad \quad \quad 5 \\
 44. & & 4 + \frac{\quad}{\quad} \\
 45. & & \quad \quad \quad 5 + \dots
 \end{array}$$

3.9) Преобразование десятичного числа X в цепную или непрерывную дробь, производится выделением целой части X, а затем целых частей

$$X_i = 1/(X_{i-1} - \text{INT}(X_{i-1}))$$

$$p = 3.1415926 \Rightarrow [3, 7, 15, 1, 252, 1, 1, 1, 4]$$

$$\begin{array}{c}
 p = 3 + \frac{1}{\frac{1}{7 + \frac{1}{15 + \frac{1}{1 + \frac{1}{252}}}}}
 \end{array}$$

3.10) Преобразование цепной дроби в обычное число

$$[3, 7, 15, 1, 252, 1, 1, 1, 4] \Rightarrow 3.1415926$$

3.11) Деление числа A на число B с произвольной точностью. (Деление в столбик)  $32/9.3 = 3.4408602150537$

Вычисляется  $F = \text{INT}(A/B)$ , а затем  $A_i = (A_{i-1} - F \cdot B) \cdot 10$

3.12) Первые европейские математики пользовались 60-ми дробями. Фибоначи приводил в качестве приблизительного решения уравнения  $x^3 + 2x^2 + 10x = 20$  значение  $1^0 22' 7'' 42''' 33^4 4^5 40^6$ . Перевести в десятичную дробь и проверить уравнение.

3.13) Вычисление числа p

2) Статистический метод

Рассмотрим четверть круга единичного радиуса и описанный квадрат.

$$X^2 + Y^2 \leq 1$$

Случайным образом выбираем точку с координатами (x,y), такими, что

$0 \leq x \leq 1$  и  $0 \leq y \leq 1$ . Если распределение случайных чисел равномерно, число k точек, попавших внутрь четверти круга, пропорционально его площади, т.е.  $\pi/4$ . Отношение  $(4k)/N$ , где N - число случайно выбранных точек, дает приближенное значение  $\pi$ .

### 3) Метод вписанных многоугольников

Длина окружности  $2\pi$ . Будем вписывать в круг многоугольники, увеличивая число их сторон так, чтобы они стремились к окружности.  $AB=2$   
Вычисляем последовательность

$$AC \quad p_1 = AC \cdot 2$$

$$AD \quad p_2 = AD \cdot 4$$

$$AE \quad p_3 = AE \cdot 8 \text{ и т.д. пока } |p_n - p_{n-1}| < \epsilon$$

4) Определить с точностью до  $\epsilon$ , определить число итераций. Печатать значение  $\pi$  при увеличении членов ряда.

$$4.1) \pi/4 = 1 - 1/3 + 1/5 - 1/7 + 1/9 \dots$$

4.2) Формула Мечина (английский математик Дж.Мечин 1680-1751)

$$\pi/4 = (4 \cdot \arctg(1/5)) - \arctg(1/239)$$

$$4.3) \pi/4 = 8 \cdot \arctg(1/10) - 4 \cdot \arctg(1/515) - \arctg(1/239)$$

$$4.4) \pi/4 = 3 \cdot \arctg(1/4) + \arctg(1/20) + \arctg(1/1985)$$

Для вычисления  $\arctg$  использовать разложение в ряд

$$\arctg(1/x) = 1/x - (1/2)x^3 + (1/5)x^5 - \dots$$

3.14) Вычисление факториала с бесконечной точностью.

Получить  $n!$ , если известно  $(n-1)!$ .

Умножаем каждое слово на n. Если в промежуточном произведении больше 6 цифр, то избыточные слева цифры надо прибавить к следующему произведению.

$$\begin{aligned} 18! &= 6 \ 402 \ 373 \ 705 \ 728 \ 000 \\ &\quad 728 \ 000 \cdot 19 = 13 \ 832 \ 000 \\ 13 + (373 \ 705 \cdot 19) &= 7 \ 100 \ 408 \end{aligned}$$

```

7+(6 402*19) = 0 121645
-----
19! = 121645100408832000

```

3.15) Вычислить число  $e$  с бесконечной точностью

$e = 1 + 1/1! + 1/2! + 1/3! + \dots + 1/n!$

3.16. Вычисление квадратного корня

Формируется прямоугольник со сторонами  $a=1$  и  $b=X$ . Площадь этого прямоугольника равна  $S = a * b = X * 1 = X$ . Преобразовав прямоугольник в квадрат так, что его площадь останется прежней, получим длину стороны, равную корню квадратному из площади фигуры, которая равна  $X$ .

Каждая итерация преобразования прямоугольника в квадрат производится следующим образом:

$S = a_0 b_0$ ;

Создаётся новый четырёхугольник, который имеет одну сторону, равную среднему арифметическому сторон текущего прямоугольника, но площадь такую же:

$S = a_1 b_1$ , где  $a_1 = (a_0 + b_0) / 2$ ,  $a b_1 = S / a_1$

$S = a_2 b_2$ , где  $a_2 = (a_1 + b_1) / 2$ ,  $a b_2 = S / a_2$

...

$S = a_n b_n$ , где  $a_n = (a_{n-1} + b_{n-1}) / 2$ ,  $a b_n = S / a_n$

И так до тех пор пока не  $|a_n - b_n| < \text{Eps}$ .

## Код функции

```

#define EPS 1e-10
float my_sqrt(float x)
{
    float S=x, a=1, b=x;
    while(fabs(a-b)< EPS){ a=(a+b)/2; b = S / a;}
    return (a+b)/2;
}

```

## *Лабораторная работа 5* *Операции с символами*

5.1) Ввести строку произвольной длины. Исключить из нее пробелы и вывести на печать.

5.2) Вести строку. Разделить ее на две равной длины и вывести на печать.

5.3) Вести строку и вывести ее в обратном порядке.

5.4) Ввести строку. Вывести на печать сначала нечетные символы, затем четные.

5.5) Ввести строку. Вывести сначала цифры, затем буквы, затем остальные символы.

5.6) Текст из строчных букв. Напечатать этот текст заглавными.

5.7) Вести строку, ввести набор символов. Если такая последовательность есть в строке заменить ее на символ \$ , если нет выдать сообщение нет повторений.

5.8) Для каждого символа текста найти число вхождений его в строку.

5.9) Определить количество различных литер в строке и выдать количество вхождений каждого символа.

5.10) Текст упорядочить по алфавиту.

5.11) Строка из слов (непрерывная последовательность букв), разделенных пробелом.

- а) Распечатать все слова по числу знаков.
- б) Распечатать все слова по алфавиту

5.12) Ввести строку символов и найти баланс открывающихся и закрывающихся скобок.

5.13) Напечатать картину, изображающую умножение в столбик

5.14) Определить

- а) количество значащих цифр в целом.
- б) сумму цифр целого числа.
- в) произведение цифр целого числа

5.15) Печать римских цифр

- 1- I      10- X

- 2-II 50-L
- 3-III 100-C
- 4-IV 500-D
- 5-V 1000-M
- 6-VI
- Если младшая цифра стоит перед старшей она вычитается IV-4
- Если младшая после старшей складывается VI -5
- Число представляется минимальным количеством знаков III-3, 4-IV, а не IIII.

4096 MMMMLXXXVI

5.16) Ввести число от 0 до 999. Распечатать его словами.

5.17) Записать целое число в обратной последовательности ( 2346 => 6432)

5.18) Определить количество трехзначных цифр сумма цифр которых равна n.

5.19) Треугольником Паскаля называется числовой треугольник

```

      1
     1 1
    1 2 1
   1 3 3 1
  1 4 6 4 1

```

Коэффициенты в разложении  $(a+b)^n$  (Бином Ньютона)

5.20)

а) Ввести посимвольно строку десятичных цифр. Выдать сообщение об ошибке, если не цифра. Перевести в вещественное число.

$$5.892 = 5 \cdot 10 + 8 \cdot 0.1 + 9 \cdot 0.01 = 2 \cdot 0.001$$

б) Ввести посимвольно строку двоичных цифр. Выдать сообщение об ошибке, если не цифра. Перевести в вещественное число.

с) Ввести посимвольно строку шестнадцатеричных цифр. Выдать сообщение об ошибке, если не цифра. Перевести в вещественное число.

5.21) Перевести из символьного в double

5.22) Перевести из double в символьный

5.23) Какая точность представления в double

5.24) а) Перевести из десятичной системы в двоичную целое

327	:	2	=	163	остаток	1
163	:	2	=	81		1
81	:	2	=	40		1
40	:	2	=	20		0
20	:	2	=	10		0
10	:	2	=	5		0
5	:	2	=	2		1
2	:	2	=	1		0
1	:	2	=	0		1

**$327_{10} = 101000111_2$**

б) Перевести из десятичной системы в шестнадцатеричную целое

7143	:	16	=	446	7
446	:	16	=	27	14 (E)
27	:	16	=	1	11 (B)
1	:	16	=	0	1

**$7143_{10} = 1BE7_{16}$**

5.25) а) Перевести из десятичной системы в двоичную дробное

0.125	x	2	=	0.250	=	0	+	0.250
0.250	x	2	=	0.5	=	0	+	0.5
0.500	x	2	=	1.000	=	1	+	0.00

**$0.125_{10} = 0.001_2$**

б) Перевести из шестнадцатеричной системы в двоичную дробное

0.2175	x	16	=	3.48	=	3	+	0.48
0.48	x	16	=	7.68	=	7	+	0.68
0.68	x	16	=	10.88	=	10 (A)	+	0.88

$$\begin{array}{rclcl}
0.88 & \times 16 & = & 14.08 & = 14 \text{ (E)} + 0.08 \\
0.08 & \times 16 & = & 1.28 & = 1 + 0.28 \\
0.28 & \times 16 & = & 4.48 & = 4 + 0.48
\end{array}$$

$$0.2175_{10} = 0.37AE14_{16}$$

б) Перевести из восьмеричной системы в двоичную дробное

с) Перевести из произвольной системы в двоичную дробное

5.26)

- а) Перевести из десятичной в двоичное любое вещественное число.
- б) Перевести из десятичной в шестнадцатеричную любое вещественное число.
- в) Перевести из десятичной в восьмеричную любое вещественное число.
- с) Перевести из десятичной в произвольную (к-ую) любое вещественное число.

5.27) Используя метод решета Эратосфена напечатать в убывающем порядке все простые число из диапазона 1-1000.

Выписываются все цифры  $> 1$ . Выбирается первое (это 2- простое) и вычеркиваются все кратные ему, кроме него самого. Затем берется следующее (3) и вычеркиваются все кратные ему.

5.28) Сколько простых чисел в каждой сотне.

5.29) Напечатать подряд все натуральные числа

012345678910111213

Найти к-ую цифру последовательности

5.30) Совершенными числами называются числа, равные сумме своих делителей ( $6=1+2+3$ ). Найти все совершенные числа от 0 до 10000 и распечатать их вместе со своими делителями.

5.31) Ввести число. Определить сумму квадратов его цифр. Повторять эту операцию с получившейся суммой и выводить результат на печать.

5.32) То же, с суммой кубов. (Если начальное число кратно трем, то всегда кончается повтором одного и того же числа)

5.33) Натуральное число из n знаков является числом Армстронга, если сумма его цифр, возведенная в n-ю степень равна самому числу.

$$153=1^3+5^3+3^3$$

Найти все четырехзначные числа Армстронга.

5.34) Возьмем любое четырехзначное число. Выпишем цифры числа в порядке возрастания и вычтем полученное число из начального. Разница будет вторым членом последовательности. Продолжая эту операцию, вскоре заметим, что "циклически" на числе 6174 независимо от начального числа.

5.35). Сколько четырехзначных чисел с неповторяющимися цифрами.

5.36) Вставить между цифрами 1,2,3,4...9 знак одной из операций  $+-*/$ , так, чтобы результат 8-операций равнялся 100. Имеется 8 решений. Распечатать их со знаками.

$$1*2+3*4*5-6+7+8-9=100$$

### ***Лабораторная работа 6. Графика***

#### ***6.1) Построить кривые по заданному параметрическому представлению***

1. Окружность радиуса  $r$  с центром в начале координат  $x=r \cos t$   $y=r \sin t$   $t \in [0, 2\pi]$
2. Эллипс с большой и малой полуосями, равными соответственно  $r_1$  и  $r_2$  и расположенными параллельно осям координат:  $x=r_1 \cos t$   $y=r_2 \sin t$   $t \in [0, 2\pi]$

#### ***6.2)***

1. Изобразить на экране точку, перемещающуюся по кругу.
2. Изобразить на экране отрезок, вращающийся вокруг своего конца.
3. Изобразить на экране расширяющуюся окружность
4. Расходящиеся окружности различного цвета

#### ***6.3) Найти кол-во целых точек, расположенных внутри окружности радиуса $R$ . (Число пикселей)***

#### ***6.4) Построить график***

1.  $\sin(x)$   $(-\pi/2, \pi/2)$  9.  $2\sin(2x)+1$   $(-\pi, \pi)$
2.  $\cos(x)$   $(0, 2\pi)$  10.  $\sin(x)+\cos(x)$   $(-\pi, \pi)$



3.  $\frac{1}{2} \sin(x) + \cos(x)$   $\frac{1}{2} (0, \pi)$  11.  $10/(1+x^2)$   $(-3, 3)$
4.  $\frac{1}{2} \sin(x) - \cos(x)$   $\frac{1}{2} (0, \pi)$  12.  $\sqrt{x^2+2}$   $(-3, 5)$
5.  $2\sin(x) + 3\cos(x)$   $(-\pi, \pi)$  13.  $(x-3)/(x^2+2)$   $(-1, 4)$
6.  $\sin(x) + \cos(2x)$   $(-\pi, \pi)$  14.  $x \cos(2x)$   $(-1, 4)$
7.  $2 - \cos(x)$   $(0, 3\pi/2)$  15.  $x^2 e^{-1/2 x^{1/2}}$   $(-1, 3)$
8.  $\sin(\sqrt{2x}) + \cos(x)$   $(0, 2\pi)$

**6.5) Построить график функции  $y = (x^2 - 2)^{2^x}$   $-2 \leq x \leq 2$**

Уточнить значения корней уравнения с точностью до  $\epsilon$ .

**6.6) Построить график функций в полярной системе координат при изменении  $z$  от 0 до  $2\pi$  с шагом 0.01 ( $R$  и  $z$  в декартовой  $R\cos(z)$ ,  $R\sin(z)$ )**

1.  $R = \sin(2z)$  четырехлепестковая роза
2.  $R = \sin(7z)$  семилепестковая роза
3.  $R = 1 + \sin(2z)$  двухлепестковая роза
4.  $R = 1 + 2\cos(z)$  улитка Паскаля
5.  $R = 1 + 2\cos(2z)$  петельное сцепление
6.  $R = az$  при  $a=1/3$   $b$   $0 \leq z < 6\pi$  с шагом 0.1

**6.7) Построить изограмму (линии одинаковых значений) функций при изменении  $x$  и  $y$**

1.  $z = 0.5(9 + x^2 - y^2 + 0.6)$   $[-3, 3]$
2.  $z = \cos(x, y)$   $[0, 10]$
3.  $z = \sin(x)\cos(x)$   $[0, 1]$
4.  $z = x - x^3/12 - y^2/2 + 0.25$   $[0, 0.1]$
5.  $z = 5 - (x+2)(x+1)(x-1)(x-2) - y^2$

**6.8) Генератор случайных чисел** генерирует случайные числа в диапазоне  $[a, b]$ . Разделим диапазон на 10 поддиапазонов одинаковой длины. Рисовать динамическую (в темпе поступления чисел) гистограмму распределения случайных чисел.

**6.9) Гистограмма массива**

**6.10) Нарисовать окружность с заданным центром и радиусом**

**6.11) Нарисовать окружность с заданным центром и радиусом и залить ее заданным цветом.**

**6.12) Нарисовать прямоугольник. Залить его заданным цветом.**

**6.13) Случайное заполнение экрана звездочками**

**6.14) Составить программу для управления размерами прямоугольника и его положением на экране.**

- > увеличивает ширину прямоугольника на 5 точек
- < уменьшает ширину прямоугольника на 5 точек
- + увеличивает высоту на 5 точек
- - уменьшает высоту на 5 точек
- Клавиши управления курсором вызывают перемещение прямоугольника в соответствующем направлении. ENTER завершает работу программы.

**6.15) Составить программу перемещения произвольного курсора по экрану. Курсор не должен оставлять след.**

**6.16) Изобразить на экране точку перемещающуюся по кругу с разной скоростью. Скорость задается > и <.**

**6.17) Нарисовать**

1. идущего человечка
2. человечка, выполняющего приседания
3. человечка, выполняющего сигнализацию флажком.
4. бегущую собачку.

**6.18) Нарисовать часы, стрелки, цифровой циферблат. Время от компьютера.**

**6.19) Рисовать (.) на окружности с некоторым шагом (большим), соединять (.) последовательно. Менять цвет.**

**6.20)**

1. Нарисовать куб в пространстве. (гексаэдр)
2. Соединить вершины через одну (тетраэдр)
3. Соединить центры тяжести (октаэдр).
4. Нарисовать икосаэдр. Две окружности 1-го радиуса параллельные оси  $xOz$  с координатами  $y +0.5$  и  $-0.5$ . Разбить каждую окрестность на 5 частей. Вершины (.) с координатой  $y +\sqrt{5}/2$  и  $-\sqrt{5}/2$ .
5. Нарисовать додекаэдр. Вершины додекаэдра центры тяжести треугольных граней икосаэдра.

### 6.30) График по сечениям.

### 6.40) Двумерный график с разных направлений.

## **Лабораторная работа 7** **Решение уравнений**

1.500 Решение уравнений методом последовательных приближений.

### Метод половинного деления

Для нахождения решения уравнения  $f(x)=0$  задается отрезок  $[a,b]$  на котором обязательно должен существовать единственный корень. Вычисляются значения функции в (.)  $a, b, x$  ( $x$ -середина отрезка) и сравниваются знаки полученных значений. Та граница, знак  $f()$  в которой совпадает со знаком  $f()$  в середине отрезка переносится в середину интервала. Операция повторяется до тех пор, пока длина отрезка не станет меньше заданной погрешности  $\epsilon$ . Корень - значение  $f()$  в середине интервала.

### Метод простых итераций

Метод простых итераций применяется к уравнению типа  $f(x)=x$ .

Для нахождения решения задается начальное приближение  $x=x^0$ , находится значение  $f(x)$ , которое берется в качестве следующего приближения  $x$ :  $x^{(k+1)}=f(x^{(k)})$ . Данная операция повторяется до тех пор, пока не выполняются условия  $|x^{(k+1)} - x^{(k)}| < \epsilon_x$  и  $|f(x^{(k+1)}) - f(x^{(k)})| < \epsilon_f$

1.510 Решить методом половинного деления. Вывести график. Определить число итераций.

Решение $[a,b]$	начальное значение $x$
-----------------	---------------------------

1	$\sin(x)=0$	$x=0$	$[-0.5,2]$	1
2	$0.1x^2 - x\ln(x) = 0$	$x=1.118$	$[1,2]$	1
3	$3\sin(\sqrt{x})+0.35x-3.8=0$	$x=1.5096$	$[3,8]$	3
4	$\sqrt{1-0.4x}-\arcsin(x)=0$	$x=0.7434$	$[0,1]$	1
5	$0.4+\arctg(\sqrt{x})-x=0$	$x=1.2388$	$[1,2]$	2
6	$e^x - e^{-x} - 2 = 0$		$[0,1]$	
7	$\arccos x - \sqrt{1-0.3x^3} = 0$	$x=0.5629$	$[0,1]$	1

1.520 Найти экстремум функции с заданной точностью  $\epsilon$  . Вывести график.

		Решение	$[a,b]$	начальное значение $x$
1	$2+x-x^2$	max 0.5	$[0,1]$	0.15
2	$(1-x)^4$	min 1.0	$[0.2,1.5]$	0.25
3	$\cos(x)+\operatorname{ch}x$	min 0	$[-0.8,0.4]$	0.25
4	$x^{1/3}(1-x)^{2/3}$	min 0.3(3)	$[0.1,0.6]$	0.1

5	$x^3 - 6x^2 + 9x + 4$	max 1.0	[0.2, 1.5]	0.3
6	$x^3 - 6x^2 + 9x + 4$	min 3.0	[2, 4]	0.3
7	$2x^2 - x^4$	nin 0.0	[-2, 0.8]	0.15
8	$\frac{x^2 - 3x + 2}{x^2 + 2x - 1}$	min 1.4	[1, 2]	0.15
9	$x^3 \ddot{O} (x-1)$	min 0.75	[0.1, 1.2]	0.2
10	$xe^{-x}$	max 1.0	[0.1, 1.5]	0.25
11	$\ln^2 x / x$	max 7.389	[6, 8]	0.15
12	$x + 1/x$	min 1.0	[0.1, 1.5]	0.2
13	$\arctg(x) - 0.5 \ln(1+x^2)$	max 1.0	[0.15, 1.5]	0.2
14	$\frac{1}{2} x^{\frac{1}{2}} e^{-\frac{1}{2} x - 1^{\frac{1}{2}}}$	max -1.0	[-2, -0.5]	0.15
15	$\ln^2 x / x$	min 1.0	[0.1, 1.9]	0.2

1.530 Решить методом итераций. Вывести график решения. Определить число итераций.

Решение    [a,b]    начальное  
значение

1	$3\sin(\sqrt{x})+0.35x-3.8=0$	[2,3]
2	$x-2+\sin(1/x)=0$	[12,2]
3	$x^2-\ln(1+x)-3$	[2,3]
4	$1-x+\sin x-\ln(1+x)=0$	[0.,1.5]
5	$x-1/(3+\sin(3.6x))=0$	[0.,0.85]
6	$\ln x-x+1.8=0$	[2,3]
7	$0.1x^2-x\ln x=0$	[1,2]
8	$x+\cos(x^{0.52}+2)=0$	[0.5,1]
9	$\sqrt{1-0.4x}-\arcsin x=0$	[0.,1.]
10	$x^2+10x-10=0$	[0.,1.]
11	$3x-4\ln x-5=0$	[2,4]
12	$0.4+\arctg(\sqrt{x})-x=0$	[1,2]

13	$\arcsin x - \sqrt{1-0.3x^3}=0$	[0.,1.]
14	$2x-3\ln x-3=0$	[0.5,0.6]
15	$x+\sqrt{x}+\sqrt[3]{x}-2.5=0$	[0.4, 1]

### *Лабораторная работа 8,9* *Работа с файлами*

8.1) Чтобы зашифровать текст, записанный с помощью русских букв и знаков препинания, его можно переписать заменив каждую букву на непосредственно следующую за не по алфавиту. (я заменяется на а)

- а) зашифровать текст
- б) расшифровать текст

8.2) Выполнить предыдущее задание при сдвиге на n букв (можно представить себе, что буквы выписаны по кругу, как цифры на циферблате)

8.3) Один из простейших способов шифровки текста состоит в табличной замене каждого символа другим символом - его шифром. Выбрать некоторую таблицу и

- а) зашифровать текст
- б) расшифровать текст

8.4) Чтобы зашифровать текст из 121 буквы, его можно записать в квадратную матрицу порядка 11 по строкам, а затем прочесть по спирали, начиная с центра.

8.5) Выбросить из программы на Си все примечания.

8.6) Программа поиска на Си элементарных синтаксических ошибок:

1. Несбалансированные скобки
2. ... с учетом примечаний

3. Двойные и одиночные кавычки.

8.7) Программа печатает в алфавитном порядке имена всех переменных в программе на Си.

8.8) Список всех слов, для каждого слова список номеров строк, где оно встречалось. (Таблица перекрестных ссылок).

8.9) Частота вхождения различных букв в текст. Нарисовать гистограмму.

8.10) В файле все русские буквы заменить заглавными.

8.11) Ввести текст. Все слова перевернуть и вывести в файл не меняя порядок слов. (abcd -> dcba))

8.12) Ввести текст из файла. Подсчитать сколько слов состоит из 1 символа, из 2-х и т.д.

8.13) Вводится слово. Выдаются все возможные комбинации из букв, входящих в это слово. Если комбинация осмыслена, то слово записывается в файл.

8.14) Перевод текста в азбуку Морзе с разной скоростью

8.15) Все маленькие буквы заменить заглавными.

8.16) Все латинские буквы заменить русскими.

### ***Лабораторная работа 10*** ***Структуры***

Распечатать в двоичном виде числа в формате

**char, int, float, double**

### ***Лабораторная работа 11*** ***Ввод-вывод***

Написать подпрограммы ввода и вывода чисел в формате **long double**.

### ***Лабораторная работа 12*** ***Классы***

1. Класс комплексных чисел



2. Класс окружностей
3. Класс прямоугольников
4. Класс векторов
5. Класс матриц
6. Класс студентов. Определить средний возраст, вес и рост.

### ***Игры***

1. "Ханойская башня". Доска имеет три колышка. На первом нанизано  $m$  дисков убывающего вверх размера. Расположить диски в том же порядке на другом колышке. Диски можно перекладывать с колышка на колышек по одному. Класть больший диск на меньший нельзя.
2. "Пятнадцать". На квадратном поле размером 4 на 4 (всего 16) расставлено 15 фишек с номерами от 1 до 15. Имеется одна свободная позиция. Расставить фишки по возрастанию их номеров. Передвигать фишки можно только на свободную позицию.
3. "Кости". Играющий называет любое число в диапазоне от 2 до 12 и ставку, которую он делает в этот ход. Программа с помощью датчика случайных чисел дважды выбирает числа от 1 до 6 (бросает кубик). Если сумма выпавших цифр меньше 7 и играющий загадал число меньше 7, он выигрывает сделанную ставку. Если сумма выпавших цифр больше 7 и задумано число больше 7, он также выигрывает ставку. Если играющий угадал сумму цифр, он получает в четыре раза больше очков, чем сделана ставка. Иначе, ставка проиграна. В начальный момент времени у играющего 100 очков.

## Паспорт экзамена

по дисциплине «Информатика», 1 семестр

### 1. Методика оценки

Экзамен проводится в письменной форме, по билетам. Билет формируется по следующему правилу: первый вопрос выбирается из диапазона вопросов 1-11, второй вопрос из диапазона вопросов 16-55 (список вопросов приведен ниже). В ходе экзамена преподаватель вправе задавать студенту дополнительные вопросы из общего перечня (п. 4).

### Форма экзаменационного билета

НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ  
Факультет АВТФ

Билет № \_\_\_\_\_  
к экзамену по дисциплине «Информатика»

---

1. Теоретический вопрос
2. Задача

Утверждаю: зав. кафедрой \_\_\_\_\_ должность, ФИО  
(подпись)  
(дата)

### 2. Критерии оценки

- Ответ на экзаменационный билет считается **неудовлетворительным**, если студент при ответе на вопросы не дает определений основных понятий, не способен показать причинно-следственные связи явлений, при решении задачи допускает принципиальные ошибки, оценка составляет 0-49 баллов.
- Ответ на экзаменационный билет засчитывается на **пороговом** уровне, если студент при ответе на вопросы дает определение основных понятий, может показать причинно-следственные связи явлений, при решении задачи допускает не принципиальные ошибки, например, вычислительные, оценка составляет 50-72 баллов.
- Ответ на экзаменационный 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 при решении задачи, оценка составляет 73-88 баллов.

- Ответ на экзаменационный 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 89-100 баллов.

### 3. Шкала оценки

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

1. Посещение лекций – 20 баллов.
2. Посещение лабораторных занятий – 20 баллов.
4. Выполнение расчетно-графического задания – 20 баллов.
5. Экзамен – 40 баллов.

### 4. Вопросы к экзамену по дисциплине «Информатика»

1. Системы счисления (Позиционные и непозиционные). Алгоритмы перевода из одной системы в другую. (Целые и дробные числа). Обратный и дополнительный код.
2. Типы данных. Переменные. Байтовые, целые, числа с плавающей запятой. Структура этих типов.
3. Операции и выражения. Перечисляемые константы. Преобразования типов (явные, неявные). Приоритеты и порядок выполнения.
4. Операторы языка. Составной оператор. Управляющие конструкции языка.
5. Условный оператор (if ... else ...). Переключатель (switch).
6. Операторы цикла (while , do while, for )
7. Операторы передачи управления (break , continue , goto , return )
8. Функции в С. Формальные и фактические параметры. Механизм передачи параметров. Возвращаемые значения. Использование указателей в качестве аргументов функции. Предварительное описание функций.
9. Адреса и указатели. Операции получения адреса и косвенная адресация. Отождествление массивов и указателей. Адресная арифметика.
10. Указатели на массивы. Массивы указателей и многомерные массивы. Динамическое выделение памяти под массивы. Инициализация указателей

11. Общая структура программы. Время существования и видимость переменных. Классы памяти
12. Структуры, объединения, перечисления
13. Рекурсивные вызовы функций. Аргументы командной строки.
14. Внешние объекты программы (extern). Статические переменные и функции (static). Регистровые переменные (register).
15. Препроцессор языка Си. (*#define #undef #include #if, #ifdef, #ifndef, #else, #elif u #endif*)
16. Написать на языке "C" или "C++" функцию, удовлетворяющую следующим требованиям, вызвать ее в программе main( )  
Прототип: int f( char \*a, char \* b );  
Параметры: a - указатель исходной строки; b - указатель результирующей строки  
**Назначение:** отсортировать символы в строке в алфавитном порядке. Алгоритм: поиск минимального в исходной строке, запись в выходную строку.
17. Написать на языке "C" или "C++" функцию, удовлетворяющую следующим требованиям, вызвать ее в программе main( )  
Прототип: void f( char \*a );  
Параметры: a - указатель исходной строки  
**Назначение:** Переставить в строке символы попарно слева и справа (1..n, 2..n-1 и т.д.);
18. Написать на языке "C" или "C++" функцию, удовлетворяющую следующим требованиям, вызвать ее в программе main( )  
Прототип: void f( char \*a );  
Параметры: a - указатель исходной строки  
**Назначение:** Отсортировать символы в строке. Алгоритм: последовательно сравниваются пары соседних элементов  $x_k$  и  $x_{k+1}$  ( $k=0, \dots, n-2$ ), если  $x_k > x_{k+1}$ , то они переставляются.
19. Написать на языке "C" или "C++" функцию, удовлетворяющую следующим требованиям, вызвать ее в программе main( )  
Прототип: void f( char \*a );  
Параметры: a - указатель исходной строки  
**Назначение:** Вывести на экран в первой строке нечетные символы, во второй четные символы исходной строки.
20. Написать на языке "C" или "C++" функцию, удовлетворяющую следующим требованиям, вызвать ее в программе main( ).  
Прототип: void f( char \*a );  
Параметры: a - указатель исходной строки  
**Назначение:** Вывести на экран сначала цифры, затем буквы, затем остальные символы исходной строки.
21. Написать на языке "C" или "C++" функцию, удовлетворяющую следующим требованиям, вызвать ее в программе main( ).  
Прототип: void f( char \*a );  
Параметры: a - указатель исходной строки  
**Назначение:** Для каждого символа исходной строки найти число вхождений его в строке и напечатать в столбец (символ, частота вхождения).
22. Написать на языке "C" или "C++" функцию, удовлетворяющую следующим требованиям, вызвать ее в программе main( )  
Прототип: void f( char \*a , char \* b );  
Параметры: a - указатель исходной строки  
b - указатель результирующей строки  
**Назначение:** Исключить из исходной строки все пробелы и записать в результирующую строку

23. Написать на языке "C" или "C++" функцию, удовлетворяющую следующим требованиям, вызвать ее в программе main( )

Прототип: int f( int \*mat, int n, int m, int &min, int &max );

Параметры: mat - указатель исходной матрицы

n – количество строк

m – количество столбцов

min – минимальное значение

max – максимальное значение

**Назначение:** Вычисление максимального и минимального значений элементов матрицы

24. Написать на языке "C" или "C++" функцию, удовлетворяющую следующим требованиям, вызвать ее в программе main( )

Прототип: int f( int \*mat, int n, int m, int & A, int & B );

Параметры: mat - указатель исходной матрицы

n – количество строк

m – количество столбцов

A – число элементов > 0

B – число элементов < 0

**Назначение:** Вычисление числа элементов больше нуля и меньше нуля

25. Написать на языке "C" или "C++" функцию, удовлетворяющую следующим требованиям, вызвать ее в программе main( )

Прототип: int f( int \*mat, int n );

Параметры: mat - указатель квадратной матрицы

n – количество строк, столбцов

m – количество столбцов

**Назначение:** Заполнение матрицы последовательностью чисел от 0 до n\*n, расположенными по спирали

1 2 3 4 5 6

20 21 22 23 24 7

19 32 33 34 25 8

18 31 36 35 26 9

17 30 29 28 27 10

16 15 14 13 12 11

26. Написать на языке "C" или "C++" функцию, удовлетворяющую следующим требованиям, вызвать ее в программе main( )

Прототип: void f( int \*A, int \*B, int n, int m );

Параметры: A - указатель исходной матрицы

B - указатель транспонированной матрицы

n – количество строк исходной матрицы

m – количество столбцов исходной матрицы

**Назначение:** Нахождение транспонированной матрицы.

Транспонирование заключается в обмене местами строк и столбцов матрицы. Элемент, расположенный в i-той строке и j-том столбце помещается в j-тую строку и i-тый столбец.

27. Написать на языке "C" или "C++" функцию, удовлетворяющую следующим требованиям, вызвать ее в программе main( )

Прототип: void f( int \*A, int \*X, int \*Y, int n, int m );

Параметры: A - указатель матрицы

X - указатель вектора (одномерного массива)

Y – указатель результирующего вектора

n – количество строк матрицы

m – количество столбцов матрицы

**Назначение:** вычисление произведения матрицы на вектор.

В результате умножения матрицы  $A[m][n]$  на вектор  $X[n]$  получается вектор  $Y[m]$ , компоненты которого вычисляются по формуле:  $y[i] = \text{SUM}(a[i][j] * x[j])$ , ( $i = 0, 2, \dots, m-1$ )

28. Написать на языке "C" или "C++" функцию, удовлетворяющую следующим требованиям, вызвать ее в программе `main()`

Прототип: `void f( int *A, int *B, int *C, int n, int m, int l );`

Параметры: A - указатель первой матрицы

B - указатель второй матрицы

C - результирующей матрицы

n – количество строк первой матрицы

m – количество столбцов первой матрицы, количество строк второй матрицы

l – количество столбцов второй матрицы

**Назначение:** вычисление произведения 2-х матриц (строка на столбец).

В результате умножения двух матриц  $A[m][n]$  и  $B[n][l]$  получается матрица  $C[m][l]$ , элементы которой находятся по следующей формуле:  $C[i][j] = \text{SUM} a[i][j] * b[k][j]$

29. Написать на языке "C" или "C++" функцию, удовлетворяющую следующим требованиям, вызвать ее в программе `main()`

Прототип: `int f( int *A, int m );`

Параметры: A - указатель квадратной матрицы

m – количество строк, столбцов матрицы

**Назначение:** вычисление среднего арифметического элементов, лежащих ниже диагонали

30. Написать на языке "C" или "C++" функцию, удовлетворяющую следующим требованиям, вызвать ее в программе `main()`

Прототип: `int f( int *a, int n );`

Параметры: a - указатель исходного массива, n - размер исходного массива.

**Назначение:** в исходном массиве все элементы, встречающиеся больше одного раза заменить на нули.

31. Написать на языке "C" или "C++" функцию, удовлетворяющую следующим требованиям, вызвать ее в программе `main()`

Прототип: `int f( int *a, int n );`

Параметры: a - указатель исходного массива, n - размер исходного массива.

**Назначение:** Отсортировать элементы массива по убыванию методом пузырька

Алгоритм : последовательно сравниваются пары соседних элементов  $x_k$  и  $x_{k+1}$  ( $k=0, n-2$ ), если  $x_k < x_{k+1}$ , то они переставляются.

32. Написать на языке "C" или "C++" функцию, удовлетворяющую следующим требованиям, вызвать ее в программе `main()`

Прототип: `int f( int *a, int n );`

Параметры: a - указатель исходного массива, n - размер исходного массива.

**Назначение:** Отсортировать элементы массива по следующему алгоритму: отыскивается максимальный элемент и переносится в конец массива, затем метод переносится ко всем элементам, кроме последнего и т.д.

33. Написать на языке "C" или "C++" функцию, удовлетворяющую следующим требованиям, вызвать ее в программе `main()`

Прототип: `int f( int M, int N );`

Параметры: M, N - целые числа

**Назначение:** Нахождение наибольшего общего делителя двух чисел N и M по рекуррентному соотношению

$N_k = N_{k-2} - \text{INT}(N_{k-2} / N_{k-1})$   $N_{k-1}$   $k=2, 3 \dots$

$N_0 = \max(|N|, |M|)$   $N_1 = \min(|N|, |M|)$

Если  $N_k = 0 \Rightarrow \text{НОД} = N_{k-1}$

( $N=23345$   $M=9135 \Rightarrow 1015$   $N=238$   $M=347 \Rightarrow 34$ )

34. Написать на языке "C" или "C++" функцию, удовлетворяющую следующим требованиям, вызвать ее в программе main( )

2.1. Прототип: int f( int \*a, int n );

2.2. Параметры: a - указатель исходного массива, n - размер исходного массива.

2.3. Назначение: в исходном массиве удаляются все элементы, значения которых меньше, чем количество их повторов в массиве. Порядок следования остальных элементов массива не изменяется.

2.4. Возвращаемое значение: количество оставшихся элементов.

2.5. Пример использования:

```
int a[] = { 1, 2, 3, 4, 2, 4, 6, 2, 4, 7, 1, 1, 5 };
```

```
int n;
```

```
printf( "n = %d\n", n = f( a, sizeof(a)/sizeof(int)));
```

```
for( int i = 0 ; i < n ; printf( "%d ", a[i++]));
```

```
// вывод на экран:
```

```
//
```

```
// n = 7
```

```
// 3 4 4 6 4 7 5
```

35. Написать на языке "C" или "C++" функцию, удовлетворяющую следующим требованиям, вызвать ее в программе main( )

2.1. Прототип: int f( int\* a, int n, int k );

2.2. Параметры: a - указатель исходного массива, n - размер массива, k - размер сдвига.

2.3. Назначение: исходный массив преобразуется в результате циклического сдвига на указанное число позиций. "Циклически" означает, что при выходе за верхнюю границу массива запись производится в позиции 0, 1 и т.д., а при выходе за нижнюю - в позиции n-1, n-2 и т.д.

2.4. Возвращаемое значение: свой первый параметр.

2.5. Пример использования:

```
int i, x[] = { 3, 6, 4, 1, 8, 2, 5, 7, 0 };
```

```
f( x, sizeof(x)/sizeof(int), -3 );
```

```
for( i = 0 ; i < sizeof(x)/sizeof(int) ; ++i ) printf( "%d ", x[i] );
```

```
f( x, sizeof(x)/sizeof(int), 2 ); printf( "\n" );
```

```
for( i = 0 ; i < sizeof(x)/sizeof(int) ; ++i ) printf( "%d ", x[i] );
```

```
// вывод на экран:
```

```
//
```

```
// 1 8 2 5 7 0 3 6 4
```

```
// 6 4 1 8 2 5 7 0 3
```

```
//
```

36. Написать на языке "C" или "C++" функцию, удовлетворяющую следующим требованиям, вызвать ее в программе main( )

2.1. Прототип: int f( char \*a, float \*b );

2.2. Параметры: a - указатель исходной строки, b - указатель буфера.

2.3. Назначение: в исходной строке ищутся и распечатываются в столбик все последовательности литер, представляющие записи целых чисел или действительных чисел в

формате <целая часть>.<дробная часть>. Сами числа в формате float последовательно записываются в буфер.

2.4. Возвращаемое значение: количество найденных и записанных чисел.

2.5. Пример использования:

```
char s[] = "12.45, 678.99, -34 0.0004 ";
```

```
float B[100];
```

```
printf( "\nЗаписано чисел: %d", f( s, B ));
```

```
// вывод на экран:
```

```
//
// 12.450000
// 678.99000
// -34.000000
// 0.000400
// Записано чисел: 4
//
```

37. Написать на языке "C" или "C++" функцию, удовлетворяющую следующим требованиям, вызвать ее в программе main( )

2.1. Прототип: int\* f( int\* a, int n );

2.2. Параметры: a - указатель исходного массива, n - размер массива.

2.3. Назначение: исходный массив преобразуется по следующему алгоритму: все нечетные элементы массива группируются в порядке возрастания в начале массива, а четные в порядке убывания в конце.

2.4. Возвращаемое значение: указатель первого четного элемента в преобразованном массиве.

2.5. Пример использования:

```
int x[] = { 3, 6, 4, 8, 2, 5, 7, 0 };
```

```
int *p = f( x, sizeof(x)/sizeof(int));
```

```
printf( "n = %d\n", p-x );
```

```
for( int i = 0 ; i < sizeof(x)/sizeof(int) ; ++i ) printf( "%d ", x[i] );
```

```
// вывод на экран:
```

```
//
```

```
// n = 3
```

```
// 3 5 7 8 6 4 2 0
```

```
//
```

38. Написать на языке "C" или "C++" функцию, удовлетворяющую следующим требованиям, вызвать ее в программе main( )

2.1. Прототип: char\* f( char \*v1, char \*v2, char \*r, int n );

2.2. Параметры: v1, v2 - указатели исходных строк, r - указатель результирующей строки, n - размер результирующей строки.

2.3. Назначение: исходные строки являются символьным представлением натуральных чисел произвольной длины. Необходимо получить в результирующей строке такое же представление **суммы** этих чисел. Не уместающиеся в результирующей строке цифры пропадают. Остающиеся свободные позиции заполняются кодом нуля.

2.4. Возвращаемое значение: указатель литеры, являющейся представлением первой значащей цифры результата.

2.5. Пример использования:

```
char a[] = "45789", b[] = "1111111", c[20];
```

```
printf( "\n%s\n%s", f( a, b, c, sizeof c ), c );
```

```
//
```

```
// вывод на экран:
```

```
//
```

```
// 1156900
```

```
// 000000000000001156900
```

```
//
```

39. Написать на языке "C" или "C++" функцию, удовлетворяющую следующим требованиям, вызвать ее в программе main( )

Прототип: double f(int n );

Назначение:  $y = 1! + 2! + 3! + 4! + \dots + n!$

40. Написать на языке "C" или "C++" функцию, удовлетворяющую следующим требованиям, вызвать ее в программе main( )



Прототип: `double f(double x, double e);`

Найти функцию с точностью  $\epsilon$ , т.е. пока  $|f_{n+1} - f_n| < \epsilon$ . Определить абсолютную разность между табличной функцией и суммой ряда. Определить с какой точностью в машине определяется эта функция.

$$e^x = 1 + x + x^2/2! + x^3/3! + \dots$$

41. Написать на языке "C" или "C++" функцию, удовлетворяющую следующим требованиям, вызвать ее в программе `main()`

Прототип: `double f(double x, double e);`

Найти функцию с точностью  $\epsilon$ , т.е. пока  $|f_{n+1} - f_n| < \epsilon$ . Определить абсолютную разность между табличной функцией и суммой ряда. Определить с какой точностью в машине определяется эта функция.

$$\sin(x) = x - x^3/3! + x^5/5! + \dots$$

42. Написать на языке "C" или "C++" функцию, удовлетворяющую следующим требованиям, вызвать ее в программе `main()`

Прототип: `double f(double x, double e);`

Найти функцию с точностью  $\epsilon$ , т.е. пока  $|f_{n+1} - f_n| < \epsilon$ . Определить абсолютную разность между табличной функцией и суммой ряда. Определить с какой точностью в машине определяется эта функция.

$$\cos(x) = 1 - x^2/2! + x^4/4! - \dots$$

43. Написать на языке "C" или "C++" функцию, удовлетворяющую следующим требованиям, вызвать ее в программе `main()`

Прототип: `double f(double x, double e);`

Найти функцию с точностью  $\epsilon$ , т.е. пока  $|f_{n+1} - f_n| < \epsilon$ . Определить абсолютную разность между табличной функцией и суммой ряда. Определить с какой точностью в машине определяется эта функция.

$$\ln(1+x) = x - x^2/2 + x^3/3 - \dots \quad |x| < 1$$

44. Написать на языке "C" или "C++" функцию, удовлетворяющую следующим требованиям, вызвать ее в программе `main()`

Прототип: `double f(double x, double e);`

Найти функцию с точностью  $\epsilon$ , т.е. пока  $|f_{n+1} - f_n| < \epsilon$ . Определить абсолютную разность между табличной функцией и суммой ряда. Определить с какой точностью в машине определяется эта функция.

$$\cos(x) = 1 - x^2/2! + x^4/4! - \dots$$

45. Написать на языке "C" или "C++" функцию, удовлетворяющую следующим требованиям, вызвать ее в программе `main()`

Прототип: `double f(double x, double e);`

Найти функцию с точностью  $\epsilon$ , т.е. пока  $|f_{n+1} - f_n| < \epsilon$ . Определить абсолютную разность между табличной функцией и суммой ряда. Определить с какой точностью в машине определяется эта функция.

$x$

$$\operatorname{tg}(x) = \frac{x}{1 - \frac{x^2}{3} + \frac{x^4}{5} - \frac{x^6}{7} + \dots}$$

$x^2$

$1 - \frac{x^2}{3}$

$x^2$

$3 - \frac{x^2}{5}$

$x^2$

$5 - \frac{x^2}{7}$

$7 - \dots$

46. Написать на языке "C" или "C++" функцию, удовлетворяющую следующим требованиям, вызвать ее в программе `main()`

Прототип: `double f(double e);`

Найти функцию с точностью  $\epsilon$ , т.е. пока  $|f_{n+1} - f_n| < \epsilon$ . Определить абсолютную разность между табличной функцией и суммой ряда. Определить с какой точностью в машине определяется эта функция.

1  
 $\text{sqrt}(2) = 1 + \frac{1}{2} + \frac{1}{8} + \frac{1}{16} + \frac{1}{64} + \dots$

47. Написать на языке "C" или "C++" функцию, удовлетворяющую следующим требованиям, вызвать ее в программе `main()`

Прототип: `double f(double e);`

Найти функцию с точностью  $\epsilon$ , т.е. пока  $|f_{n+1} - f_n| < \epsilon$ . Определить абсолютную разность между табличной функцией и суммой ряда. Определить с какой точностью в машине определяется эта функция.

2  
 $e = 2 + \frac{1}{3} + \frac{1}{4} + \frac{1}{5} + \frac{1}{6} + \dots$

48. Написать на языке "C" или "C++" функцию, удовлетворяющую следующим требованиям, вызвать ее в программе `main()`

Прототип: `double f(double x, double e);`

Найти функцию с точностью  $\epsilon$ , т.е. пока  $|f_{n+1} - f_n| < \epsilon$ . Определить абсолютную разность между табличной функцией и суммой ряда. Определить с какой точностью в машине определяется эта функция.

49. Написать на языке "C" или "C++" функцию, удовлетворяющую следующим требованиям, вызвать ее в программе `main()`

Прототип: `void f(int n);`

Печать римских цифр

1. 1- I 10-X
2. 2-II 50-L
3. 3-III 100-C
4. 4-IV 500-D
5. 5-V 1000-M
6. 6-VI

Соблюдая следующие правила:

7. Если младшая цифра стоит перед старшей она вычитается IV-4
8. Если младшая цифра стоит после старшей - складывается VI -5
9. Число представляется минимальным количеством знаков III-3, 4-IV, а не IIII.

50. Написать на языке "C" или "C++" функцию, удовлетворяющую следующим требованиям, вызвать ее в программе `main()`

Прототип: `int f(int n, int k);`

Напечатать подряд все натуральные числа до  $n$  - 012345678910111213 Найти  $k$ -ую цифру последовательности

51. Написать на языке "C" или "C++" функцию, удовлетворяющую следующим требованиям

Найти все четырехзначные числа Армстронга. Натуральное число из  $n$  знаков является числом Армстронга, если сумма его цифр, возведенная в  $n$ -ю степень равна самому числу.  $153=1^3+5^3+3^3$

52. Написать на языке "C" или "C++" функцию, удовлетворяющую следующим требованиям

Вставить между цифрами 1,2,3,4...9 знак одной из операций  $+,-,*,/$ , так, чтобы результат 8-операций равнялся 100. Имеется 8 решений. Распечатать их со знаками.

$$1*2+3*4*5-6+7+8-9=10$$

53. Написать на языке "C" или "C++" функцию, удовлетворяющую следующим требованиям

Совершенными числами называются числа, равные сумме своих делителей ( $6=1+2+3$ ).

Найти все совершенные числа от 0 до 10000 и распечатать их вместе со своими делителями.

54. Написать на языке "C" или "C++" функцию, удовлетворяющую следующим требованиям, вызвать ее в программе main( )

Прототип: void f1(int n, \*char a); // a – массив, состоящий из 0 или 1. В main( ) массив распечатать

Перевести из десятичной системы в двоичную целое число

$$327 : 2 = 163 \text{ остаток } 1$$

$$163 : 2 = 81 \text{ } 1$$

$$81 : 2 = 40 \text{ } 1$$

$$40 : 2 = 20 \text{ } 0$$

$$20 : 2 = 10 \text{ } 0$$

$$10 : 2 = 5 \text{ } 0$$

$$5 : 2 = 2 \text{ } 1$$

$$2 : 2 = 1 \text{ } 0$$

$$1 : 2 = 0 \text{ } 1$$

$$\mathbf{32710 = 1010001112}$$

55. Написать на языке "C" или "C++" функцию, удовлетворяющую следующим требованиям, вызвать ее в программе main( )

Прототип: void f1(double b, \*char a); // a – массив, состоящий из 0 или 1. В main( ) массив распечатать

Перевести из десятичной системы в двоичную дробное

$$0.125 \times 2 = 0.250 = 0 + 0.250$$

$$0.250 \times 2 = 0.5 = 0 + 0.5$$

$$0.500 \times 2 = 1.000 = 1 + 0.00$$

$$\mathbf{0.12510 = 0.0012\_}$$

## **Паспорт расчетно-графического задания (работы)**

по дисциплине «Информатика», 1 семестр

### **1. Методика оценки**

В рамках расчетно-графического задания (работы) по дисциплине студенты ознакомиться с типами данных в языке C#.

При выполнении расчетно-графического задания (работы) студенты должны рассмотреть основные типы данных и написать код для выполнения некоторых операций с числами.

Обязательные структурные части РГЗ.

1. Оглавление.
2. Задание.
3. Ответы на задания.

Оцениваемые позиции:

1. Качество кода.

### **2. Критерии оценки**

- Работа считается выполненной **на пороговом** уровне, если части РГЗ(Р) выполнены формально, оценка составляет 50-72 баллов.
- Работа считается выполненной **на базовом** уровне, если имеются некоторые ошибки при ответе на вопросы, оценка составляет 73-88 баллов.
- Работа считается выполненной **на продвинутом** уровне, если работа выполнена полностью и без ошибок, оценка составляет 89-100 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.

1. Посещение лекций – 20 баллов.
2. Посещение лабораторных занятий – 20 баллов.
3. Выполнение расчетно-графического задания – 20 баллов.
4. Экзамен – 40 баллов.

### **4. Примерный перечень тем РГЗ(Р)**

1. Описать структуру представления чисел в форматах char, int, float, double.
2. Привести примеры представления чисел в форматах char, int, float, double.
3. Написать программу на языке C, которая должна

- a. В двоичном виде представить число в формате char, int, float, double. (Для этого использовать объединение union)
- b. Для чисел типа float разбить на три целых слова: знак, характеристика, мантисса. Определить порядок.
- c. Для чисел типа float перевести в десятичный вид и вывести на печать отдельно целую часть и отдельно дробную.
- d. Дробную часть переводить из двоичной формы в десятичную по правилам перевода умножением на 10.