

Министерство образования и науки Российской Федерации  
ФГБОУ ВПО «Ивановский государственный университет»

**Г.С. Степович-Цветкова**

**А.В. Розов**

**СБОРНИК ЗАДАЧ  
ПО ЯЗЫКАМ И ТЕХНОЛОГИЯМ  
ПРОГРАММИРОВАНИЯ**

*для студентов бакалавриата  
направлений "Математика",  
"Математика и компьютерные науки",  
"Фундаментальная информатика и информационные  
технологии",  
"Информационная безопасность"*

Иваново  
Издательство «Ивановский государственный университет»  
2016

УДК 681.3.06  
ББК 32.973.26-018.1я73  
367

**Степович-Цветкова Г.С., Розов А.В.**

Сборник задач по языкам и технологиям программирования : задачник для студентов бакалавриата направлений "Математика", "Математика и компьютерные науки", "Фундаментальная информатика и информационные технологии", "Информационная безопасность" / Г.С. Степович-Цветкова, А.В. Розов. — Электрон. дан. — Иваново : Иван. гос. ун-т, 2016. — 1 электрон. опт. диск (CD-ROM) ; 12 см. — Систем. требования: программа чтения файлов в формате PDF 1.4. — ISBN 978-5-7807-1143-8.

Издание содержит задачи для проведения практических и лабораторных занятий по курсам Язык и технологии программирования и Язык программирования, а также справочный материал по основам программирования на языке C++, классам и динамическим структурам данных.

Предназначено студентам бакалавриата направлений "Математика", "Математика и компьютерные науки", "Фундаментальная информатика и информационные технологии", "Информационная безопасность".

*Печатается по решению редакционно-издательского совета  
Ивановского государственного университета*

*Рецензенты:*

кафедра физики, математики и информатики ГБОУ ВПО  
«Ивановская государственная медицинская академия»

Министерства здравоохранения РФ

(зав. кафедрой доктор технических наук,  
профессор **Е.В. Березина**)

кандидат технических наук **Е.А. Шварев**

(ФГБОУ ВО «Ивановская пожарно-спасательная академия»  
ГПС МЧС России)

ISBN 978-5-7807-1143-8 © Степович-Цветкова Г.С.,  
Розов А.В., 2016

© ФГБОУ ВПО «Ивановский  
государственный университет», 2016

## Оглавление

1.	Структурное программирование на языке С++.....	4
1.1.	Основные элементы языка программирования С++ .....	4
1.1.1.	Реализация линейных алгоритмов. Ввод и вывод информации.....	4
1.1.2.	Реализация алгоритмов ветвления .....	5
1.1.3.	Реализация циклических алгоритмов.....	6
1.1.4.	Одномерные и многомерные статические и динамические массивы.....	7
1.1.5.	Структуры, перечисления .....	12
1.2.	Разработка функций пользователя.....	14
1.2.1.	Определение функций пользователя.....	14
1.2.2.	Перегрузка операторов и функций.....	17
1.2.3.	Рекурсивные алгоритмы .....	19
1.2.4.	Библиотеки функций и типов данных. Многомодульные приложения .....	20
1.3.	Библиотеки функций С++ .....	23
1.3.1.	Строковые операции .....	23
1.3.3.	Файловый потоковый ввод-вывод.....	25
1.3.4.	Математические функции.....	29
2.	Объектно-ориентированное программирование .....	31
2.1.	Классы.....	31
2.1.1.	Инкапсуляция .....	31
2.1.2.	Полиморфизм.....	35
2.1.3.	Наследование .....	37
2.1.4.	Шаблоны классов .....	38
2.2.	Динамические структуры данных.....	42
2.2.1.	Массив с контролем индексов .....	42
2.2.2.	Список .....	44
2.2.3.	Стек. Реализация с помощью списка и массива ..	47
2.2.4.	Очередь. Реализация с помощью списка и массива	48
2.2.5.	Дерево.....	49
	Справочный материал.....	51
	Список литературы .....	73

# 1. Структурное программирование на языке C++

## 1.1. Основные элементы языка программирования C++

### 1.1.1. Реализация линейных алгоритмов. Ввод и вывод информации

1. Вычислите сумму и разность двух вещественных чисел, введенных с клавиатуры. Результат выведите на экран.
2. Вычислите среднее арифметическое трех целых чисел, заданных с клавиатуры. Результат выведите на экран.
3. Вычислите периметр и площадь треугольника, заданного длинами его сторон (значения введите с клавиатуры). Результат выведите на экран.
4. Вычислите площадь и гипотенузу прямоугольного треугольника по заданным катетам (значения введите с клавиатуры). Результат выведите на экран.
5. Вычислите радиус вписанной окружности прямоугольного треугольника по заданным гипотенузе и одному катету (значения введите с клавиатуры). Результат выведите на экран.
6. Вычислите объем шара по заданному радиусу (величину радиуса шара введите с клавиатуры). Результат выведите на экран.
7. Вычислите объем куба и площадь всей его поверхности по заданной длине ребра куба, значение которой введите с клавиатуры. Результат выведите на экран.
8. Вычислите значение функции  $y = 2x^2 - \frac{10}{x}$  при  $x$ , значение которого вводится с клавиатуры. Результат выведите на экран.

9. Вычислите значение выражения  $\frac{2a+5b^3}{15a^2}$ , если  $a$  и  $b$  вводятся с клавиатуры ( $a \neq 0$ ). Результат выведите на экран.
10. Вычислите значение функции  $z = \frac{2 \cdot (x^3 + 2) + x \cdot (y^3 + 3)}{6 + 5 \cdot y^2 + x^2}$ , значения  $x$  и  $y$  введите с клавиатуры. Результат выведите на экран.

### 1.1.2. Реализация алгоритмов ветвления

11. Определите знак заданного целого числа. Если оно является положительным, то увеличьте его втрое. Если оно отрицательно, то уменьшите на единицу. Если число равно нулю, то замените его на 55. Результат выведите на экран.
12. Выведите на экран строку-описание целого числа, заданного в диапазоне от 1 до 999, вида «четное двузначное число», «нечетное трехзначное число» и т. д.
13. Найдите наибольшее значение из трех заданных действительных чисел  $a$ ,  $b$  и  $c$ .
14. Найдите наименьшее значение из квадратов двух заданных действительных чисел  $a$  и  $b$ .
15. Определите, принадлежит ли точка  $A(x,y)$  окружности с радиусом  $R$ .
16. Определите в какой четверти декартовой системы координат располагается точка  $A(x,y)$ .
17. Проверьте два треугольника, заданных длинами сторон, на равенство.
18. Вычислите значение функции
- $$y = \begin{cases} \frac{1}{x^2 - 7x + 10}, & \text{если } x \in (-\infty, 0) \cup (7, +\infty), \\ x^2 - 7x + 10, & \text{в противном случае} \end{cases}$$
- по заданному вещественнозначному значению  $x$ .

19. Вычислите значение функции  

$$z = \begin{cases} 2 \cdot (x + y^2), & \text{если } x > 0 \text{ и } y > 0, \\ x^2 - y, & \text{в противном случае} \end{cases}$$
 по заданным вещественнозначным значениям  $x$  и  $y$ .

20. Вычислите значение функции  

$$z = \begin{cases} 0, & \text{если } x \leq 0 \text{ и } y \leq 0, \\ x^2 + y^2, & \text{если } x + y > 0, \\ \frac{1}{x + y}, & \text{в противном случае} \end{cases}$$
 по заданным вещественнозначным значениям  $x$  и  $y$ .

### 1.1.3. Реализация циклических алгоритмов

21. По заданной последовательности: 2, 9, 37, 149, 597, ... ( $a_n = 4 \cdot a_{n-1} + 1$ ) и заданному числу  $n$ , вычислите сумму первых  $n$  элементов последовательности.
22. По заданной последовательности: 2, 5, 26, 677, ... ( $a_n = a_{n-1}^2 + 1$ ) вычислите сумму элементов последовательности, меньших 10000.
23. Даны последовательность чисел  $1/2, 1/9, 1/28, \dots, 1/(n^3+1), \dots$  и  $x$  – достаточно малое положительное число. Найдите порядковый номер и значение члена этой последовательности, меньшего, чем  $x$ .
24. Даны последовательность чисел  $2, 9, 28, \dots, (n^3+1), \dots$  и число  $x$ . Найдите порядковый номер и значение первого члена этой последовательности, большего  $x$ .
25. Даны действительное число  $a$  и число  $n$ . Вычислите  $a(a-n)(a-2n)(a-3n)\dots(a-jn)$ , где  $|a-(j-1)n| < 1000 \leq |a-jn|$ . Если  $j$  с требуемыми свойствами не существует, выведите соответствующее сообщение.
26. Дано действительное число  $a$  ( $a > 1$ ). Вычислите  $a^n$  такое, что  $a^n \leq 100 < a^{n+1}$

27. Дано действительное число  $a$  ( $0 < a < 1$ ). Вычислите  $a^n$  такое, что  $a^{n-1} > 0,0005 > a^n$
28. Вычислите  $a(n) = \sqrt{3 + \sqrt{3 + \dots \sqrt{3}}}$  ( $n$  корней), где  $n$  такое, что  $a(n-1) < 2,3017 < a(n)$ .
29. По заданным натуральному числу  $n$  и вещественному  $x$  вычислите  $\sum_{i=1}^n \left(x + \frac{i}{x}\right)$ .
30. По заданным натуральному числу  $n$  и вещественному  $x$  вычислите  $\sum_{i=1}^n (-1)^i x^{i+1}$ .
31. По заданным натуральному числу  $n$  и вещественному  $x$  вычислите  $\prod_{i=1}^n \frac{x^2}{x^2 + 2}$ .
32. По заданным натуральному числу  $n$  и вещественному  $x$  вычислите  $\prod_{i=1}^n \left(1 + \frac{x}{i!}\right)$ .
33. В заданной последовательности из  $n$  целочисленных элементов определите количество четных чисел.
34. В заданной последовательности из  $n$  вещественных элементов определите количество неотрицательных чисел.
35. В заданной последовательности из  $n$  целочисленных элементов найдите минимальное и максимальное значение.

#### 1.1.4. Одномерные и многомерные статические и динамические массивы

36. Дан целочисленный массив с  $n$  элементами. В массиве каждый элемент, модуль которого при делении на  $p$  дает в остатке  $q$ , замените нулем.
37. Дан целочисленный массив с  $n$  элементами. В заданном массиве определите сумму минимального и максимального элементов.

38. Дан целочисленный массив с  $n$  элементами. Получите удвоенную сумму всех положительных членов последовательности. Если таких элементов в последовательности нет, выведите соответствующее сообщение.
39. Дан вещественнозначный массив с  $n$  элементами. Найдите количество тех элементов, которые больше нуля, но меньше единицы.
40. Дан вещественный массив с  $n$  элементами. Вычислите сумму элементов массива, расположенных между первым и вторым отрицательными элементами. Если таких элементов в последовательности нет, выведите соответствующее сообщение.
41. Дан вещественный массив с  $n$  элементами. Вычислите сумму модулей элементов массива, расположенных после первого элемента, равного нулю. Если таких элементов в последовательности нет, выведите соответствующее сообщение.
42. Дан вещественный массив с  $n$  элементами. Вычислите произведение элементов массива, расположенных после максимального по модулю элемента. Если таких элементов в последовательности нет, выведите соответствующее сообщение.
43. Дан вещественный массив с  $n$  элементами. Вычислите сумму модулей элементов массива, расположенных после минимального по модулю элемента. Если таких элементов в последовательности нет, выведите соответствующее сообщение.
44. Дан массив символов произвольной длины. Определите количество символов, не являющихся цифрами.
45. Дан массив символов произвольной длины. Замените всюду один заданный символ на другой заданный символ.
46. Дан массив символов произвольной длины. Удалите из массива все пробелы.
47. Дан массив символов произвольной длины. Замените каждую букву английского алфавита на следующую по алфавиту букву (последняя буква алфавита заменяется на первую).



48. Дан массив символов произвольной длины. Найдите количество различных символов.
49. Дан массив символов произвольной длины. Вставьте после каждого символа заданный символ.
50. Дан массив символов произвольной длины. Проверьте соблюдение баланса открывающихся и закрывающихся круглых скобок.
51. Дан массив символов произвольной длины. Замените каждую цифру на следующую по порядку цифру (цифра 9 заменяется на цифру 0).
52. Дан массив символов произвольной длины. Замените каждую последовательность из двух и более пробелов на один пробел.
53. Дан массив символов произвольной длины. Замените всюду один заданный символ на последовательность из пяти таких символов.
54. Дан массив символов произвольной длины. Найдите количество вхождений заданного символа в строке.
55. Дан массив символов произвольной длины. Замените каждую цифру, большую пяти, символом \*.
56. Дан вещественный двумерный массив размерности  $n \times m$ . Определите номер строки, сумма элементов которой максимальна.
57. Дан вещественный двумерный массив размерности  $n \times m$ . Замените нулевые элементы на максимальный элемент матрицы.
58. Дан вещественный двумерный массив размерности  $n \times m$ . Замените элементы, стоящие на главной диагонали, на минимальный элемент матрицы.
59. Дан вещественный двумерный массив размерности  $n \times m$ . Вычислите произведение элементов последнего столбца и первой строки матрицы.
60. Дан вещественный двумерный массив размерности  $n \times m$ . Вычислите сумму элементов первой и последней строк матрицы.
61. Дан вещественный двумерный массив размерности  $n \times m$ . Вычислите произведение первого и последнего столбцов матрицы.

62. Дан вещественный двумерный массив размерности  $n \times m$ . Выведите на экран минимальные в строках элементы матрицы.
63. Дан вещественный двумерный массив размерности  $n \times m$ . Определите номер столбца, произведение элементов которого минимально.
64. Дан вещественный двумерный массив размерности  $n \times m$ . Замените все четные элементы матрицы средним арифметическим элементов последнего столбца.
65. Дан вещественный двумерный массив размерности  $n \times m$ . Поменяйте местами первый и последний столбцы матрицы.
66. Дан вещественный двумерный массив размерности  $n \times m$ . Замените все отрицательные элементы матрицы суммой положительных элементов матрицы.
67. Дан вещественный двумерный массив размерности  $n \times m$ . Выведите на экран сумму максимальных элементов столбцов матрицы.
68. Дан вещественный двумерный массив размерности  $n \times m$ . Замените все нечетные элементы матрицы количеством нулевых элементов данной матрицы.
69. Дан вещественный двумерный массив размерности  $n \times m$ . Поменяйте местами первую строку с той строкой, в которой расположен минимальный элемент матрицы.
70. Дан вещественный двумерный массив размерности  $n \times m$ . Вычислите произведение элементов тех строк, в которых расположены минимальный и максимальный элементы массива.
71. Разработайте алгоритм обработки двумерного динамического массива размера  $n \times m$ . Необходимо выбрать в каждой строке минимальный элемент, а затем из минимумов выбрать максимальный.
72. Разработайте алгоритм обработки двумерного динамического массива размера  $n \times m$ , заполненного целыми случайными числами из диапазона от  $a$  до  $b$ . Необходимо каждый элемент строки умножить на максимальный. В результате необходимо вывести на экран первоначальный массив, все максимумы и полученный вновь массив.

73. Разработайте алгоритм обработки двумерного динамического массива размера  $n \times m$ , заполненного целыми случайными числами из диапазона от  $a$  до  $b$ . Необходимо каждый элемент массива, меньший диагонального в строке заменить на среднеарифметическое диагональных элементов. Также должна быть проверка, что  $n \leq m$ . В результате необходимо вывести на экран первоначальный массив и полученный вновь массив.
74. Разработайте алгоритм обработки двумерного динамического массива размера  $n \times m$ , заполненного целыми случайными числами из диапазона от  $a$  до  $b$ . Необходимо к каждому элементу массива прибавить следующий элемент (к последнему элементу добавить неизменный первый).
75. Разработайте алгоритм обработки двумерного динамического массива размера  $n \times m$ , заполненного целыми случайными числами из диапазона от  $a$  до  $b$ . Необходимо каждую строку заменить суммой двух последующих строк (предпоследняя строка заменяется на сумму первой (неизменной) и последней, последняя строка заменяется на сумму первой (неизменной) и второй (неизменной)). В результате необходимо вывести на экран первоначальный массив и полученный вновь массив.
76. Разработайте алгоритм обработки двумерного динамического массива размера  $n \times m$ , заполненного целыми случайными числами из диапазона от  $a$  до  $b$ . Необходимо определить произведение всех положительных элементов. В результате необходимо вывести на экран первоначальный массив и полученное произведение.
77. Разработайте алгоритм обработки двумерного динамического массива размера  $n \times m$ , заполненного целыми случайными числами из диапазона от  $a$  до  $b$ . Необходимо определить индексы минимального и максимального элементов массива.
78. Разработайте алгоритм обработки двумерного динамического массива размера  $n \times m$ , заполненного целыми случайными числами из диапазона от  $a$  до  $b$ . Необходимо определить сумму всех элементов, у которых первый индекс не больше второго. В результате необходимо вывести на экран первоначальный массив и полученную сумму.

79. Разработайте алгоритм обработки двумерного динамического массива размера  $n \times m$ , заполненного целыми случайными числами из диапазона от  $a$  до  $b$ . Необходимо разместить элементы строк в обратном порядке. В результате необходимо вывести на экран первоначальный массив и полученный массив.
80. Разработайте алгоритм обработки двумерного динамического массива размера  $n \times m$ . Необходимо выбрать в каждом столбце максимальный элемент. Результаты (все максимумы с индексами) выведите на экран.
81. Разработайте алгоритм обработки двумерного динамического массива размера  $n \times m$ . Необходимо каждый элемент массива увеличить на сумму элементов его строки.
82. Разработайте алгоритм обработки двумерного динамического массива размера  $n \times m$ . Необходимо определить индексы минимального элемента из максимальных по строкам элементов массива.

### 1.1.5. Структуры, перечисления

83. Разработайте алгоритм, вычисляющий пересекаются ли две окружности. Точки и окружности задайте с помощью структур.
84. Разработайте алгоритм, вычисляющий длину отрезка и середину отрезка, заключенного между двумя точками. Точки задайте в виде структур.
85. Разработайте алгоритм, вычисляющий сумму и разность двух векторов в трехмерном пространстве, а также умножение вектора на скаляр. Вектора задайте в виде структур.
86. Разработайте алгоритм, вычисляющий пересекаются ли окружность и прямая. Точки, прямую и окружность задайте с помощью структур.
87. Разработайте алгоритм, вычисляющий сумму, разность, произведение и частное двух комплексных чисел. Комплексные числа задайте в виде структуры.

88. Разработайте алгоритм, вычисляющий скалярное, векторное и смешанное произведение векторов в трехмерном пространстве. Вектора задайте с помощью структуры.
89. Разработайте алгоритм, вычисляющий точку пересечения двух прямых. Точки и прямые задайте с помощью структур.
90. Разработайте алгоритм, вычисляющий площадь поверхности, объем и диагональ параллелепипеда, заданного вершинами. Вершины параллелепипеда задайте с помощью структуры.
91. Разработайте алгоритм, вычисляющий периметр, и площадь треугольника, заданного вершинами. Вершины треугольника задайте с помощью структуры.
92. Разработайте алгоритм, определяющий принадлежит ли точка плоскости. Точку и плоскость задайте в виде структур.
93. Разработайте алгоритм, определяющий взаимное расположение двух прямых в пространстве. Прямые задайте с помощью структур.
94. Разработайте алгоритм, определяющий взаимное расположение прямой и плоскости в пространстве. Прямую и плоскость задайте с помощью структур.
95. Напишите программу вывода на экран значения заданной цифры прописью (если введена цифра 9, на экране появится «девять»), тип переменной для хранения цифр задайте перечислением.
96. Напишите программу вывода на экран названия дня недели по заданному порядковому номеру (1 – понедельник и т.д.), тип переменной для хранения номера дня недели задайте перечислением.
97. С помощью перечисления задайте тип «дни недели». Реализуйте алгоритм определения, является ли заданный день выходным или нет.
98. Напишите программу вывода на экран названия месяца по заданному порядковому номеру, тип переменной для хранения номера месяца задайте перечислением.
99. Напишите программу перевода некоторой заданной суммы из одной валюты в другую, набор доступных валют задайте перечислением, организуйте возможность выбора валют для перевода, состав валют и их курс выберите произвольно.

## 1.2. Разработка функций пользователя

### 1.2.1. Определение функций пользователя

100. Напишите функцию, которая выбирает из трех элементов минимальный, и вызовите ее в основном теле программы.
101. Напишите функцию, которая меняет местами значения двух переданных ей элементов, и организуйте ее вызов в основном теле программы.
102. Напишите функцию решения линейного уравнения вида  $ax+b=0$  и организуйте ее вызов в основном теле программы.
103. Напишите функцию для вычисления длины отрезка на плоскости по заданным точкам, ограничивающим этот отрезок, и организуйте ее вызов в основном теле программы.
104. Напишите функцию для вычисления площади треугольника, заданного координатами его вершин на плоскости, и вызовите ее в основном теле программы.
105. Напишите функцию для вычисления модуля векторного произведения двух векторов, заданных координатами на плоскости, и организуйте ее вызов в основном теле программы.
106. Напишите функцию для вычисления скалярного произведения двух векторов, заданных координатами в трехмерном пространстве, и организуйте ее вызов в основном теле программы.
107. Напишите функцию для вычисления длины вектора в трехмерном пространстве по заданным координатам, и организуйте ее вызов в основном теле программы.
108. Вычислите значение величины, содержащей несколько однотипных сумм. Для вычисления сумм напишите функцию

$$\text{пользователя. } y = \frac{6 + a \cdot \sum_{i=2}^m (3 \cdot i^2 + 2 \cdot i + c)}{4 + \sum_{i=1}^m (i^2 + 2) + \sqrt{\sum_{i=2}^m (i + 3)}}$$

109. Вычислите значение величины, содержащей несколько однотипных сумм. Для вычисления сумм напишите функцию

$$\text{пользователя. } y = \frac{2 \cdot \sum_{k=1}^m (k^3 + 2) + a \cdot \sum_{k=1}^m (k^3 + 3)}{6 + \sum_{k=3}^{m+2} (5 \cdot k^3 + a)}.$$

110. Вычислите значение величины, содержащей несколько однотипных сумм. Для вычисления сумм напишите функцию

$$\text{пользователя. } y = \frac{3 \cdot \sum_{i=3}^m (i + 2) + a \cdot \sum_{i=4}^{m+1} (i^2 + 2)}{a + \sum_{i=5}^{m+2} (2 \cdot i^2 + i + 4)}.$$

111. Вычислите значение величины, содержащей несколько однотипных сумм. Для вычисления сумм напишите функцию

$$\text{пользователя. } y = \frac{2 \cdot \sum_{i=2}^m (3 \cdot i^3 + 5)}{2 \cdot \sum_{i=2}^m (3 \cdot i^2 + i + 1) + \sum_{i=3}^{m+1} (2 \cdot i^3 + b)}.$$

112. Вычислите значение величины, содержащей несколько однотипных сумм. Для вычисления сумм напишите функцию

$$\text{пользователя. } y = \frac{a \cdot \sum_{k=3}^n (2 \cdot k + 1)^2}{1 + 2 \cdot \sum_{k=1}^m (3 \cdot k + a)^2 + 3 \cdot \sum_{k=3}^n (k + 3)^2}.$$

113. Вычислите значение величины, содержащей несколько однотипных сумм. Для вычисления сумм напишите функцию

$$\text{пользователя. } y = \frac{b + 2 \cdot \sum_{j=2}^m (j^2 + 2)}{1 + 3 \cdot \sum_{j=3}^n (2 \cdot j^2 + 3) + 4 \cdot \sum_{j=2}^m (3 \cdot j^2 + j + 4)}.$$

114. Вычислите значение величины, содержащей несколько однотипных сумм. Для вычисления сумм напишите функцию

$$\text{пользователя. } y = \frac{a + \sum_{k=1}^m (k^3 + 3) + 3 \cdot \sum_{k=3}^n (k^2 + 3)}{10 + \sum_{j=1}^m (2 \cdot j^3 + j^2 + 1)}.$$

115. Вычислите значение величины, содержащей несколько однотипных сумм. Для вычисления сумм напишите функцию

$$\text{пользователя. } y = \frac{5 + 3 \cdot \sum_{k=3}^n (2 \cdot k^2 + 1)}{1 + \sum_{k=2}^m (k^2 + k + 2) + 4 \cdot \sum_{i=3}^n (5 \cdot k + 3)}.$$

116. Вычислите значение величины, содержащей несколько однотипных сумм. Для вычисления сумм напишите функцию

$$\text{пользователя. } y = \frac{3 \cdot \sum_{i=2}^n (i^2 + 2) + 2 \cdot \sum_{j=3}^m (j^3 + 1)}{2 + \sum_{k=3}^m (2 \cdot k^3 + k + 2)}.$$

117. Напишите функцию, которая вычисляет произведение отрицательных элементов одномерного массива.

118. Напишите функцию, которая вычисляет сумму элементов одномерного массива, начиная с заданного элемента (элемент задан индексом, который пользователь вводит с клавиатуры).

119. Напишите функцию, которая вычисляет произведение элементов одномерного массива, которые больше заданного числа (число пользователь вводит с клавиатуры).

120. Напишите функцию, которая находит сумму четных элементов одномерного массива.

121. Напишите функцию, определяющую является ли элемент одномерного массива четным, положительным числом.

122. Напишите функцию, которая вычисляет произведение с первого элемента двумерного массива до заданного элемента массива (элемент задан двумя индексами, которые пользователь вводит с клавиатуры).



123. Напишите функцию, сортирующую каждую строку двумерного массива в порядке убывания.
124. Напишите функцию, которая вычисляет сумму элементов двумерного массива, меньших заданного числа (число пользователь вводит с клавиатуры).
125. Напишите функцию, сортирующую каждый столбец двумерного массива в порядке возрастания.
126. Напишите функцию, которая вычисляет произведение элементов двумерного массива, являющихся делителем заданного числа.

### 1.2.2. Перегрузка операторов и функций

127. Реализуйте операции сравнения для типа Вектор, заданного структурой.
128. Реализуйте арифметические операции для комплексных чисел, заданных структурой.
129. Реализуйте арифметические операции для рациональных чисел, заданных структурой.
130. Реализуйте операции сравнения для типа Треугольник, заданного структурой.
131. Реализуйте операции инкремента и декремента для рациональных чисел, заданных структурой.
132. Реализуйте операции инкремента и декремента для комплексных чисел, заданных структурой.
133. Напишите функции для нахождения максимального элемента из двух, из трех чисел и в одномерном массиве.
134. Напишите функцию вычисления суммы элементов одномерного массива, больших произвольной величины  $a$ . Реализуйте перегрузку этой функции для случая вычисления аналогичной суммы элементов в двумерном массиве, в которой сумму элементов, больших произвольной величины  $a$ , нужно просчитать для каждого столбца, используя первую функцию.
135. Напишите перегруженные функции для обработки одномерного и двумерного массивов. Необходимо вычислить

- значения наименьшего элемента и его индекс: в одномерном массиве среди всех элементов, а в двумерном – для каждой строки.
136. Напишите перегруженные функции для обработки одномерного и двумерного массивов. Необходимо вычислить среднее значение: в одномерном массиве среди всех элементов, а в двумерном – для каждой строки.
137. Напишите перегруженные функции для обработки одномерного и двумерного массивов. Необходимо вычислить среднее отклонение элементов от среднего значения: в одномерном массиве среди всех элементов, а в двумерном – для каждого столбца.
138. Напишите перегруженные функции для обработки одномерного и двумерного массивов. Необходимо вычислить значение разности между наибольшим и наименьшим элементами: в одномерном массиве для всех элементов, а в двумерном – для каждого столбца.
139. Напишите перегруженные функции для обработки одномерного и двумерного массивов. В одномерном массиве необходимо поменять местами максимальный и минимальный элементы. В двумерном массиве необходимо вычислить сумму элементов для каждого столбца и поменять местами столбцы с максимальным и минимальным значением суммы.
140. Напишите перегруженные функции для обработки одномерного и двумерного массивов. В одномерном массиве необходимо поменять местами максимальный и минимальный элементы. В двумерном массиве необходимо вычислить произведение элементов для каждой строки. Поменять местами столбцы с максимальным и минимальным значением произведения.
141. Напишите перегруженные функции для обработки одномерного и двумерного массивов. Необходимо выполнить нормировку элементов: в одномерном массиве, поделив все элементы на максимальный, а в двумерном массиве элементы каждой строки матрицы необходимо поделить на значение максимального элемента по строке.

142. Напишите перегруженные функции для обработки одномерного и двумерного массивов. Необходимо выполнить нормировку элементов: в одномерном массиве, поделив все элементы на максимальный, а в двумерном массиве элементы каждого столбца матрицы необходимо поделить на значение максимального элемента по столбцу.

### 1.2.3. Рекурсивные алгоритмы

143. Напишите рекурсивную функцию для подсчета факториала заданного целого числа.

144. Напишите рекурсивную функцию для возведения в целую степень произвольного целого числа по формуле  $a^n = a \cdot a^{n-1}$ .

145. Напишите рекурсивную функцию для вычисления  $n$ -го члена арифметической прогрессии  $a_n = a_{n-1} + 5$  при заданном  $a_1$ .

146. Напишите рекурсивную функцию для вычисления  $n$ -го члена геометрической прогрессии  $b_n = 3b_{n-1}$  при заданном  $b_1$ .

147. Напишите рекурсивную функцию для вычисления  $n$ -го члена последовательности чисел Фибоначчи:  $f_n = f_{n-1} + f_{n-2}$ ,  $f_0 = 1$ ,  $f_1 = 1$ .

148. Напишите рекурсивную функцию для вычисления  $n$ -го члена последовательности:  $a_n = 2a_{n-1} + 3$ ,  $a_1$  вводится с клавиатуры.

149. Напишите рекурсивную функцию для вычисления  $n$ -го члена последовательности:  $a_n = 2a_{n-1} + 3a_{n-2}$ ,  $a_0$  и  $a_1$  вводятся с клавиатуры.

150. Напишите рекурсивную функцию для вычисления  $n$ -го члена последовательности:  $a_n = 2a_{n-1} + 3a_{n-2} - a_{n-3}$ ,  $a_0$ ,  $a_1$  и  $a_2$  вводятся с клавиатуры.

151. Напишите рекурсивную функцию для вычисления суммы  $n$  первых членов арифметической прогрессии  $a_n = a_{n-1} - 3$  при заданном  $a_1$ .
152. Напишите рекурсивную функцию для вычисления суммы  $n$  первых членов геометрической прогрессии  $b_n = -4b_{n-1}$  при заданном  $b_1$ .
153. Напишите рекурсивную функцию для вычисления суммы  $n$  членов последовательности:  $a_n = 3a_{n-1} - a_{n-2}$ , начиная с элемента под номером  $k$ ,  $a_0$  и  $a_1$  вводятся с клавиатуры.
154. Напишите рекурсивную функцию для вычисления среднего арифметического первых  $n$  членов последовательности:  $a_n = \frac{1}{3}a_{n-1} + 7$ ,  $a_1$  вводится с клавиатуры.
155. Напишите рекурсивную функцию для вычисления произведения первых  $n$  членов последовательности:  $a_n = -4a_{n-1} + \frac{2}{5}$ ,  $a_1$  вводится с клавиатуры.
156. Напишите рекурсивную функцию для вычисления произведения  $n$  членов последовательности:  $a_n = -5a_{n-1} + 3a_{n-2}$ , начиная с элемента под номером  $k$ ,  $a_0$  и  $a_1$  вводятся с клавиатуры.

#### 1.2.4. Библиотеки функций и типов данных. Многомодульные приложения

157. Создайте библиотеку, содержащую тип данных Вектор. Реализуйте операции сравнения, сложения, вычитания, скалярного произведения, функции нахождения длины вектора, векторного произведения и вывода вектора на экран. Используя созданную библиотеку, вычислите  $a \cdot (b + c)$ , координаты вектора  $(a \times b) - c$  и его длину, определите, какой из векторов имеет наибольшую длину (с помощью

- операций сравнения). Координаты векторов  $a$ ,  $b$  и  $c$  введите с клавиатуры.
158. Создайте библиотеку, содержащую тип данных Комплексное число, заданное в алгебраической форме. Реализуйте операции сложения, вычитания, умножения, деления, функции нахождения модуля комплексного числа и вывода комплексного числа на экран. Используя созданную библиотеку, вычислите  $\frac{a \cdot (b + c)}{d - b}$  и определите модули всех заданных комплексных чисел. Действительные и мнимые части комплексных чисел  $a$ ,  $b$ ,  $c$  и  $d$  введите с клавиатуры.
159. Создайте библиотеку, содержащую тип данных Комплексное число, заданное в тригонометрической форме. Реализуйте операции сложения, вычитания, умножения, деления, функции возведения комплексного числа в  $n$ -ю степень, извлечения корня  $n$ -ой степени и вывода комплексного числа на экран. Используя созданную библиотеку, вычислите  $\frac{a^5 \cdot (b + c)^9}{\sqrt[3]{d - b}}$ . Модули и аргументы комплексных чисел  $a$ ,  $b$ ,  $c$  и  $d$  введите с клавиатуры.
160. Создайте библиотеку, содержащую тип данных Комплексное число, заданное в показательной форме. Реализуйте операции сложения, вычитания, умножения, деления, функции перевода числа в алгебраическую и тригонометрическую формы и вывода комплексного числа, заданного в показательной, алгебраической или тригонометрической формах, на экран. Используя созданную библиотеку, вычислите  $\frac{a \cdot (b + c)}{d - b}$ , результат выведите в показательной, алгебраической и тригонометрической формах. Модули и аргументы комплексных чисел  $a$ ,  $b$ ,  $c$  и  $d$  введите с клавиатуры.
161. Создайте библиотеку, содержащую тип данных Рациональное число. Реализуйте операции сложения, деления, функции сокращения рационального числа, нахождения обратной дроби и вывода рационального числа

- на экран. Используя созданную библиотеку, определите число  $\frac{a+b+c}{d+b}$ , при необходимости выполнив сокращение, и обратную дробь для этого числа, результат выведите на экран. Числители и знаменатели рациональных чисел  $a$ ,  $b$ ,  $c$  и  $d$  введите с клавиатуры.
162. Создайте библиотеку, содержащую тип данных Рациональное число. Реализуйте операции вычитания, умножения, сравнения, функции возведения в степень и вывода рационального числа на экран в виде обыкновенной дроби и десятичного числа. Используя созданную библиотеку, сравните числа  $(a-b)^9$  и  $c^3 \cdot (a-d)$ , полученные числа и результат сравнения выведите на экран. Числители и знаменатели рациональных чисел  $a$ ,  $b$ ,  $c$  и  $d$  введите с клавиатуры.
163. Создайте библиотеку, содержащую тип данных Матрица, заданная размерностью и массивом элементов. Реализуйте операции сложения, вычитания, умножения двух матриц, функции умножения матрицы на число, вычисления определителя матрицы и вывода матрицы на экран. Используя созданную библиотеку, найдите матрицу  $3A + B \cdot (C - A)$  и вычислите ее определитель. Элементы матриц  $A$ ,  $B$  и  $C$  с учетом необходимых размерностей введите с клавиатуры.
164. Создайте библиотеку, содержащую тип данных Матрица, заданная размерностью и массивом элементов. Реализуйте операции сложения, вычитания, умножения двух матриц, функции определения, является ли матрица нулевой, единичной, диагональной, и вывода матрицы на экран. Используя созданную библиотеку, определите, является ли матрица  $A \cdot (B + C - A)$  нулевой, единичной или диагональной, определите, является ли матрица  $B$  обратной матрице  $A$ . Элементы матриц  $A$ ,  $B$  и  $C$  с учетом необходимых размерностей введите с клавиатуры.
165. Создайте библиотеку, содержащую тип данных Матрица, заданная размерностью и массивом элементов. Реализуйте

операции сложения, вычитания, умножения двух матриц, функции определения, является ли матрица симметричной, верхнетреугольной и нижнетреугольной, и вывода матрицы на экран. Используя созданную библиотеку, найдите матрицу  $(A + B) \cdot (C - D)$  и определите, является ли она симметричной, верхнетреугольной или нижнетреугольной. Элементы матриц  $A$ ,  $B$ ,  $C$  и  $D$  с учетом необходимых размерностей введите с клавиатуры.

166. Создайте библиотеку, содержащую тип данных Дата, заданная днем, месяцем и годом. Реализуйте операции сравнения двух дат, функции нахождения периода времени между двумя датами, определения времени года по дате и вывода даты на экран. Используя созданную библиотеку, определите какое из событий  $a$  или  $b$  произошло раньше, сколько времени прошло от события  $a$  до события  $b$ , в какое время года произошли события  $a$  и  $b$ . Даты  $a$  и  $b$  введите с клавиатуры.

### **1.3. Библиотеки функций C++**

#### **1.3.1. Строковые операции**

167. Дана строка. Определите, сколько в ней содержится цифр.
168. Дана строка. Удалите из нее все цифры.
169. Дана строка. Замените в ней все строчные английские буквы на прописные.
170. Дана строка. Замените в ней все идущие подряд пробелы одним пробелом. Удалите начальные и завершающие пробелы.
171. Дана строка. Вставьте в начало строки слово “Tomato”, если в ней не встречается ни одной буквы “t” (регистр не учитывается).
172. Дана строка. Замените в ней каждое слово “Alice” словом “Bob”.

173. Дана строка. Замените в ней каждое слово “Bob” словом “Alice”.
174. Дана строка, содержащая несколько слов, разделенных пробелами и знаками препинания. Подсчитайте количество слов и найдите самое длинное слово.
175. Дана строка, содержащая несколько слов, разделенных пробелами и знаками препинания. Поменяйте местами самое длинное и самое короткое слово.
176. Дана строка. Вставьте после каждого символа “.” пробел в случае его отсутствия.
177. Дана строка, содержащая запись натурального числа. Вставьте запятые после каждой группы из трех цифр, начиная с правого символа (например, “1000000” перепишется как “1,000,000”).
178. Дана строка и натуральное число  $n$ . Получите строку, в которой  $n$  раз повторяется исходная строка.
179. Даны две строки:  $s_1$  и  $s_2$ . Определите, сколько раз  $s_1$  содержится в  $s_2$ .
180. Даны две строки:  $s_1$  и  $s_2$ . Удалите из  $s_2$  первое вхождение  $s_1$ .
181. Дана строка, содержащая представление числа  $x$  в десятичной системе счисления. Запишите это число в целочисленную переменную.
182. Дана строка, содержащая представление числа  $x$  в шестнадцатеричной системе счисления. Переведите  $x$  в десятичную систему счисления.
183. Два целых числа записаны с помощью строк. Сложите числа, и запишите результат в третью строку.
184. Дана строка, содержащая арифметическое выражение, составленное из чисел, знаков операций и скобок. Проверьте правильность расстановки скобок.
185. Дана строка, содержащая несколько слов. Выведите на экран первые буквы всех слов, содержащих более двух символов.
186. Дана строка, содержащая несколько слов. Выведите на экран все слова, начинающиеся и заканчивающиеся на одну и ту же букву. Регистр не учитывается.



187. Дана строка, содержащая несколько слов. Выведите на экран все слова, в которых буква “t” встречается не менее двух раз.
188. Дана строка, содержащая несколько слов. Выведите на экран все слова, в которых нет повторяющихся букв.
189. Дана строка, содержащая несколько слов. Выведите на экран все слова в порядке от последнего к первому.
190. Дана строка, содержащая несколько слов. Выведите на экран все слова, получающиеся с помощью перестановок символов слова “abcde”.
191. Дана строка, содержащая несколько слов. Найдите в ней два слова, одно из которых получается с помощью перестановки символов другого.
192. Дана строка, содержащая несколько слов. Найдите слово, встречающееся наибольшее количество раз.
193. Дана строка, содержащая несколько слов, разделенных пробелами. Переставьте слова в порядке возрастания их длин.
194. Даны две строки s1 и s2, каждая из которых содержит несколько слов, разделенных пробелами. Запишите в третью строку через пробел все слова, которые входят как в s1, так и в s2.

### **1.3.3. Файловый потоковый ввод-вывод**

195. Дан текстовый файл. Создайте файл, в который перепишите содержимое исходного файла, добавляя четыре пробела в начало каждого абзаца.
196. Дан текстовый файл. Проверьте, все ли предложения в файле начинаются с прописной буквы. Если нет, внесите соответствующие изменения.
197. Дан текстовый файл. Перепишите его содержимое в новый файл, оставляя только буквы английского алфавита, заменяя все строчные буквы прописными.
198. Дан текстовый файл. Удалите все символы:
  - а) не являющиеся цифрами и буквами английского алфавита;
  - б) не являющиеся цифрами и буквами русского алфавита.

199. Дан текстовый файл. Подсчитайте общее количество строк в файле и количество строк максимальной длины. Соответствующие результаты допишите в конец файла.
200. Дан текстовый файл, содержащий синтаксически корректный текст программы на языке C++. Проверьте правильность расстановки скобок “(”, “)” и “{”, “}”.
201. Дан текстовый файл, содержащий синтаксически корректный текст программы на языке C++. Удалите из программы все комментарии.
202. В текстовом файле записаны целые числа. Пользователь вводит с клавиатуры число. Проверьте, содержится ли в файле заданное число. Соответствующие результаты допишите в конец файла (если заданное число содержалось в файле, укажите «координаты» его первого вхождения: номер строки и позицию в ней).
203. В текстовом файле записаны целые числа. Найдите сумму и количество чётных чисел файла. Соответствующие результаты допишите в конец файла.
204. В текстовом файле записаны целые числа. Найдите номера строк, содержащие минимальный и максимальный элементы. В строке, содержащей минимальный элемент, найдите произведение и количество ненулевых элементов. Соответствующие результаты допишите в конец файла (если минимальный или максимальный элемент несколько раз встречаются в файле, то учитываются только их первые вхождения).
205. В текстовом файле записаны вещественные числа. Найдите среднее арифметическое положительных элементов файла и «координаты» (номер строки и позицию в ней) последнего вхождения максимального элемента. Соответствующие результаты запишите в новый текстовый файл.
206. В текстовом файле записаны вещественные числа. Замените в каждой строке элементы с наибольшим значением нулём. Создайте файл, который будет содержать данные исходного файла, а затем данные преобразованного файла.
207. В текстовом файле записаны вещественные числа. Пользователь вводит с клавиатуры число. Подсчитайте

- количество чисел в каждой строке. Найдите номера строк, содержащих заданное пользователем число. Соответствующие результаты запишите в новый текстовый файл.
208. В текстовом файле записаны вещественные числа. Найдите количество положительных чисел и их среднее арифметическое значение. Затем удалите первую строку с минимальным количеством чисел в строке.
209. В текстовом файле записаны целые числа. Пользователь вводит с клавиатуры число. Переместите из файла все числа, не превосходящие заданного числа, в новый файл с сохранением их расположения по строкам. Если в текстовом файле таких чисел нет, дополните его строкой с заданным числом.
210. В текстовом файле записаны целые числа. Из каждой строки файла удалите числа, предшествующие первому положительному числу. Если в строке нет положительных чисел, строка остается без изменений.
211. В многострочном текстовом файле записаны слова, разделённые пробелами. В каждой строке файла найдите длину самого короткого слова и подсчитайте количество слов данной длины в этой строке. Результаты обработки строки поместите в конец каждой строки.
212. В текстовом файле, содержащем  $n$  строк, записаны слова, разделённые пробелами. Создайте  $n$ -строчный текстовый файл, в каждую строку которого перепишите первое слово из соответствующей строки в исходном файле. Удалите первые слова из каждой строки исходного файла.
213. В многострочном текстовом файле записаны слова, разделённые пробелами. Запишите перед каждым словом его номер в строке. В начало каждой строки запишите ее номер.
214. Дан текстовый файл. Пользователь вводит с клавиатуры число  $n$ . Перепишите в новый текстовый файл те строки исходного файла, длина которых не меньше  $n$ .
215. В многострочном текстовом файле записаны слова, разделённые пробелами. Найдите количество пустых строк и количество строк, которые начинаются и оканчиваются на

- одну и ту же букву. Соответствующие результаты допишите в конец файла.
216. Дан текстовый файл. Пользователь вводит с клавиатуры число. Проверьте, содержит ли файл строку заданной длины. Если в файле имеются строки заданной длины, то в первой из них измените порядок следования символов на обратный.
  217. Дан текстовый файл. Вставляя в начало каждой строки необходимое число пробелов, выровняйте его содержимое по правому краю и перепишите в новый текстовый файл.
  218. Дан текстовый файл. Все чётные строки этого файла запишите во второй файл, а нечётные – в третий. Порядок следования строк сохраняется.
  219. Дан текстовый файл. Перепишите его содержимое (кроме символов перевода строки) в другой текстовый файл, вставляя символы перевода строки после каждой точки, а также через 40 символов после предыдущего перевода строки.
  220. Даны два текстовых файла. Подсчитайте количество символов в каждом из них и, в случае совпадения полученных значений, сравните файлы на поэлементное равенство. На экран выведите соответствующее сообщение о равенстве или неравенстве длин файлов. В последнем случае укажите номер строки и позицию в ней первого символа, отличающегося в файлах.
  221. Дан текстовый файл. Пользователь вводит с клавиатуры числа  $b$  и  $e$ . Перепишите в новый текстовый файл строки с номерами из отрезка  $[b;e]$  (если это возможно, в противном случае выведите соответствующее текстовое сообщение на экран).
  222. Дан текстовый файл. Замените подряд идущие пробелы одним пробелом и уберите пробелы, с которых начинаются строки.
  223. Дан текстовый файл. Измените в нём порядок следования символов на обратный.
  224. Дан текстовый файл, состоящий из нескольких строк. Поменяйте в нём местами первую и последнюю строку.

225. Даны два текстовых файла. В первом из них записаны слова, разделённые пробелами. Для каждого из этих слов подсчитайте, сколько раз оно встречается во втором файле.

### 1.3.4. Математические функции

226. Предусмотрев ввод исходных данных с клавиатуры или из файла, вычислите значение выражения:  $\frac{\sin \alpha + \sin 3\alpha + \sin 5\alpha}{\cos 5\beta - \cos 3\beta}$ . Результат выведите на экран или в файл.

227. Предусмотрев ввод исходных данных с клавиатуры или из файла, вычислите значение выражения:  $\sqrt{2} \operatorname{tg} \alpha \cdot \sin^3 \left( \frac{\alpha + \beta}{2} \right)$ .

Результат выведите на экран или в файл.

228. Предусмотрев ввод исходных данных с клавиатуры или из файла, вычислите значение выражения:  $\frac{\arcsin^2(\alpha + 3\alpha + 5\alpha)}{\arcsin^3(\beta + 5\beta + 9\beta)}$ .

229. Предусмотрев ввод исходных данных с клавиатуры или из файла, вычислите значение выражения:  $\frac{1 - \arccos|\alpha + \beta|}{1 + \arccos \alpha}$ .

Результат выведите на экран или в файл.

230. Предусмотрев ввод исходных данных с клавиатуры или из файла, вычислите значение выражения:  $\sqrt{\frac{|1 - x - x^3 - x^5|}{x^4 + 16}}$ .

Результат выведите на экран или в файл.

231. Предусмотрев ввод исходных данных с клавиатуры или из файла, вычислите значение выражения:  $\sum_{i=1}^{15} \operatorname{arctg} \left( \frac{\alpha + 2\beta}{i} \right)$ .

Результат выведите на экран или в файл.

232. Вычислите значение выражения:  $\sum_{i=1}^{50} \left| 1 - i + \frac{\sqrt{i}}{5} \right|$ . Результат выведите на экран или в файл.
233. Предусмотрев ввод исходных данных с клавиатуры или из файла, вычислите значение выражения:  $e^{5x^2/(3x-1)}$ . Результат выведите на экран или в файл.
234. Предусмотрев ввод исходных данных с клавиатуры или из файла, вычислите значение выражения:  $\ln 15 \cdot e^{\sqrt{1+\sin^2 x + \sin^4 x}}$ . Результат выведите на экран или в файл.
235. Предусмотрев ввод исходных данных с клавиатуры или из файла, вычислите значение выражения:  $\lg \left( \frac{a^2 + b^2}{5|a|} \right)$ . Результат выведите на экран или в файл.
236. Предусмотрев ввод исходных данных с клавиатуры или из файла, вычислите значение выражения:  $\ln \left| \lg \left| a + b^2 + \frac{(a-b)^{10}}{5e^{15}} \right| \right|$ . Результат выведите на экран или в файл.
237. Сгенерируйте последовательность из  $n$  случайных значений. Вычислите сумму квадратов этих значений. Результат выведите на экран или в файл.
238. Сгенерируйте последовательность из  $n$  случайных значений. Вычислите разность максимального и минимального элементов последовательности. Результат выведите на экран или в файл.
239. Сгенерируйте последовательность из  $n$  случайных значений. Вычислите среднее арифметическое элементов последовательности, округленное до ближайшего целого в большую сторону. Результат выведите на экран или в файл.
240. Сгенерируйте последовательность из  $n$  случайных значений. Вычислите квадратный корень из произведения элементов, меньших первого. Результат необходимо

округлить до ближайшего целого в меньшую сторону и вывести на экран или в файл.

## 2. Объектно-ориентированное программирование

### 2.1. Классы

#### 2.1.1. Инкапсуляция

241. Определите класс Точка, заданная  $n$  координатами. Реализуйте конструктор, деструктор, метод вывода объекта класса на экран, метод определения расстояния от точки до начала координат. Создайте в классе статическую переменную для хранения количества созданных экземпляров класса и статические методы для увеличения и уменьшения значения статической переменной и вывода на экран информации о количестве созданных экземпляров класса.
242. Определите класс Комплексное число, заданное действительной и мнимой частью. Реализуйте конструктор, деструктор, метод вывода объекта класса на экран, метод определения модуля комплексного числа, а также два дополнительных метода, добавленных разработчиком. Создайте в классе статическую переменную для хранения количества созданных экземпляров класса и статические методы для увеличения и уменьшения значения статической переменной и вывода на экран информации о количестве созданных экземпляров класса.
243. Определите класс Рациональное число, заданное числителем и знаменателем. Реализуйте конструктор, деструктор, метод вывода объекта класса на экран, метод

- сокращения дробей, а также два дополнительных метода, добавленных разработчиком. Создайте в классе статическую переменную для хранения количества созданных экземпляров класса и статические методы для увеличения и уменьшения значения статической переменной и вывода на экран информации о количестве созданных экземпляров класса.
244. Определите класс Матрица размера  $n \times m$ . Реализуйте конструктор, деструктор, метод вывода объекта класса на экран, метод транспонирования матрицы, а также два дополнительных метода, добавленных разработчиком. Создайте в классе статическую переменную для хранения количества созданных экземпляров класса и статические методы для увеличения и уменьшения значения статической переменной и вывода на экран информации о количестве созданных экземпляров класса.
245. Определите класс Окружность, заданная радиусом и координатами центра. Реализуйте конструктор, деструктор, метод вывода объекта класса на экран, метод определения, пересекаются ли две заданные окружности, а также два дополнительных метода, добавленных разработчиком. Создайте в классе статическую переменную для хранения количества созданных экземпляров класса и статические методы для увеличения и уменьшения значения статической переменной и вывода на экран информации о количестве созданных экземпляров класса.
246. Определите класс Трапеция, заданная длинами оснований и углами при основании. Реализуйте конструктор, деструктор, метод вывода объекта класса на экран, метод определения периметра заданной трапеции, а также два дополнительных метода, добавленных разработчиком. Создайте в классе статическую переменную для хранения количества созданных экземпляров класса и статические методы для увеличения и уменьшения значения статической переменной и вывода на экран информации о количестве созданных экземпляров класса.
247. Определите класс Ромб, заданный длиной стороны и углом между двумя сторонами. Реализуйте конструктор, деструктор, метод вывода объекта класса на экран, метод



определения площади заданного ромба, а также два дополнительных метода, добавленных разработчиком. Создайте в классе статическую переменную для хранения количества созданных экземпляров класса и статические методы для увеличения и уменьшения значения статической переменной и вывода на экран информации о количестве созданных экземпляров класса.

248. Определите класс Эллипс, заданный двумя полуосями. Реализуйте конструктор, деструктор, метод вывода объекта класса на экран, метод определения площади заданного эллипса, а также два дополнительных метода, добавленных разработчиком. Создайте в классе статическую переменную для хранения количества созданных экземпляров класса и статические методы для увеличения и уменьшения значения статической переменной и вывода на экран информации о количестве созданных экземпляров класса.

249. Определите класс Параллелограмм, заданный координатами вершин. Реализуйте конструктор, деструктор, метод вывода объекта класса на экран, метод вычисления площади параллелограмма, метод проверки, имеют ли два параллелограмма общие точки, а также два дополнительных метода, добавленных разработчиком. Создайте в классе статическую переменную для хранения количества созданных экземпляров класса и статические методы для увеличения и уменьшения значения статической переменной и вывода на экран информации о количестве созданных экземпляров класса.

250. Определите класс Треугольник, заданный координатами вершин. Реализуйте конструктор, деструктор, метод вывода объекта класса на экран, метод вычисления площади треугольника и проверки, имеют ли два треугольника общие точки, а также два дополнительных метода, добавленных разработчиком. Создайте в классе статическую переменную для хранения количества созданных экземпляров класса и статические методы для увеличения и уменьшения значения статической переменной и вывода на экран информации о количестве созданных экземпляров класса.

251. Определите класс Прямоугольник, заданный координатами вершин. Реализуйте конструктор, деструктор, метод вывода объекта класса на экран, метод вычисления периметра прямоугольника и проверки, имеют ли два прямоугольника общие точки, а также два дополнительных метода, добавленных разработчиком. Создайте в классе статическую переменную для хранения количества созданных экземпляров класса и статические методы для увеличения и уменьшения значения статической переменной и вывода на экран информации о количестве созданных экземпляров класса.
252. Определите класс Сфера в трехмерном пространстве, заданная координатами центра и радиусом. Реализуйте конструктор, деструктор, метод вывода объекта класса на экран, метод вычисления объема сферы и метод, проверяющий пересекаются ли две сферы, а также два дополнительных метода, добавленных разработчиком. Создайте в классе статическую переменную для хранения количества созданных экземпляров класса и статические методы для увеличения и уменьшения значения статической переменной и вывода на экран информации о количестве созданных экземпляров класса.
253. Определите класс Прямая, заданная координатами точки и направляющего вектора. Реализуйте конструктор, деструктор, метод вывода объекта класса на экран, метод проверки пересекаются ли две прямые и нахождения расстояния от заданной точки до прямой, а также два дополнительных метода, добавленных разработчиком. Создайте в классе статическую переменную для хранения количества созданных экземпляров класса и статические методы для увеличения и уменьшения значения статической переменной и вывода на экран информации о количестве созданных экземпляров класса.
254. Определите класс, описывающий число вида  $c \cdot x^a$ , где  $x$  – произвольное число. Реализуйте конструктор, деструктор, метод вывода объекта класса на экран, метод, реализующий вычисление данного числа при заданном  $x$ , а также два

дополнительных метода, добавленных разработчиком. Создайте в классе статическую переменную для хранения количества созданных экземпляров класса и статические методы для увеличения и уменьшения значения статической переменной и вывода на экран информации о количестве созданных экземпляров класса.

255. Определите класс, описывающий число вида  $a + b \cdot e^x$ , где  $x$  – произвольное число. Реализуйте конструктор, деструктор, метод вывода объекта класса на экран, метод, реализующий вычисление данного числа при заданном  $x$ , а также два дополнительных метода, добавленных разработчиком. Создайте в классе статическую переменную для хранения количества созданных экземпляров класса и статические методы для увеличения и уменьшения значения статической переменной и вывода на экран информации о количестве созданных экземпляров класса.

## 2.1.2. Полиморфизм

256. В классе Точка, заданной  $n$  координатами, реализуйте конструктор по умолчанию и конструктор, задающий поля равными переданным в качестве параметров значениям. Реализуйте перегрузку операций сравнения, присваивания, вывода на экран, индексирования.
257. В классе Комплексное число, заданное действительной и мнимой частью реализуйте конструктор по умолчанию и конструктор, задающий поля равными переданным в качестве параметров значениям. Реализуйте перегрузку арифметических операций, операций сравнения, присваивания, вывода на экран.
258. В классе Рациональное число, заданное числителем и знаменателем реализуйте конструктор по умолчанию и конструктор, задающий поля равными переданным в качестве параметров значениям. Реализуйте перегрузку арифметических операций, операций сравнения, присваивания, вывода на экран.

259. В классе Матрица размера  $n \times m$  реализуйте конструктор по умолчанию и конструктор, задающий поля равными переданным в качестве параметров значениям. Реализуйте перегрузку арифметических операций, операций сравнения, присваивания, вывода на экран, индексирования.
260. В классе Окружность, заданная радиусом и координатами центра, реализуйте конструктор по умолчанию и конструктор, задающий поля равными переданным в качестве параметров значениям. Реализуйте перегрузку операций сравнения, присваивания, вывода на экран.
261. В классе Трапеция, заданная длинами оснований и углами при основании, реализуйте конструктор по умолчанию и конструктор, задающий поля равными переданным в качестве параметров значениям. Реализуйте перегрузку операций сравнения, присваивания, вывода на экран.
262. В классе Ромб, заданный длиной стороны и углом между двумя сторонами, реализуйте конструктор по умолчанию и конструктор, задающий поля равными переданным в качестве параметров значениям. Реализуйте перегрузку операций сравнения, присваивания, вывода на экран.
263. В классе Эллипс, заданный двумя полуосями, реализуйте конструктор по умолчанию и конструктор, задающий поля равными переданным в качестве параметров значениям. Реализуйте перегрузку операций сравнения, присваивания, вывода на экран.
264. В классе Параллелограмм, заданный координатами вершин, реализуйте конструктор по умолчанию и конструктор, задающий поля равными переданным в качестве параметров значениям. Реализуйте перегрузку операций сравнения, присваивания, вывода на экран.
265. В классе Треугольник, заданный координатами вершин, реализуйте конструктор по умолчанию и конструктор, задающий поля равными переданным в качестве параметров значениям. Реализуйте перегрузку операций сравнения, присваивания, вывода на экран.
266. В классе Прямоугольник, заданный координатами вершин, реализуйте конструктор по умолчанию и конструктор, задающий поля равными переданным в качестве

параметров значениям. Реализуйте перегрузку операций сравнения, присваивания, вывода на экран.

267. В классе Сфера в трехмерном пространстве, заданная координатами центра и радиусом, реализуйте конструктор по умолчанию и конструктор, задающий поля равными переданным в качестве параметров значениям. Реализуйте перегрузку операций сравнения, присваивания, вывода на экран.

268. В классе Прямая, заданная координатами точки и направляющего вектора, реализуйте конструктор по умолчанию и конструктор, задающий поля равными переданным в качестве параметров значениям. Реализуйте перегрузку операций сравнения, присваивания, вывода на экран.

269. В классе, описывающем число вида  $c \cdot x^a$ , где  $x$  – произвольное число, реализуйте конструктор по умолчанию и конструктор, задающий поля равными переданным в качестве параметров значениям. Реализуйте перегрузку арифметических операций, операций сравнения, присваивания, вывода на экран.

270. В классе, описывающем число вида  $a + b \cdot e^x$ , где  $x$  – произвольное число. Реализуйте конструктор по умолчанию и конструктор, задающий поля равными переданным в качестве параметров значениям. Реализуйте перегрузку арифметических операций, операций сравнения, присваивания, вывода на экран.

### 2.1.3. Наследование

271. Определите базовый класс «n-угольник». Затем от него определите два производных класса «прямоугольник» и «квадрат», в каждом из которых реализуйте заданный метод: площадь.

272. Определите базовый класс «фигура». Затем от него определите два производных класса «окружность» и «шар», в каждом из которых реализуйте заданный метод: площадь.

273. Определите базовый класс «точка». Затем от него определите два производных класса «прямоугольник» и «параллелепипед», в каждом из которых реализуйте заданный метод: площадь.
274. Определите базовый класс «многочлен». Затем от него определите два производных класса «квадратный многочлен» и «кубический многочлен», в каждом из которых реализуйте заданный метод: вычисление значения многочлена в точке.
275. Определите базовый класс «уравнение». Затем от него определите два производных класса «квадратное уравнение» вида  $ax^2 + bx + c = 0$  и «кубическое уравнение» вида  $x^3 + px + q = 0$ , в каждом из которых реализуйте заданный метод: нахождение корней.
276. Определите базовый класс «матрица». Затем от него определите два производных класса «матрица 2x2» и «матрица 3x3», в каждом из которых реализуйте заданный метод: определитель.
277. Определите базовый класс «n-угольник». Затем от него определите два производных класса «ромб» и «квадрат», в каждом из которых реализуйте заданный метод: площадь.
278. Определите базовый класс «многогранник». Затем от него определите два производных класса «октаэдр» и «додекаэдр», в каждом из которых реализуйте заданный метод: объем.
279. Определите базовый класс «n-мерный вектор». Затем от него определите два производных класса «2-мерный вектор» и «3-мерный вектор», в каждом из которых реализуйте заданный метод: норма.

#### 2.1.4. Шаблоны классов

280. Создайте шаблон класса для работы с векторами в  $n$ -мерном пространстве. Параметр-тип шаблона должен определять тип данных для компонент вектора, параметр-константа – размерность вектора. Реализуйте 2 конструктора (инициализации и копирования), деструктор, метод *read* ввода компонент вектора с клавиатуры, метод нахождения

длины вектора, угла между двумя векторами. Перегрузите операции “=” присваивания, “+” и “\*” суммы и скалярного произведения двух векторов и “[ ]” индексации. Продемонстрируйте работу всех реализованных методов для векторов трех- и четырехмерного пространства с компонентами типа `float` и `unsigned short` соответственно.

281. Создайте шаблон класса для работы с многочленами одной переменной. Параметр-тип шаблона должен определять тип данных для коэффициентов многочлена. Реализуйте 2 конструктора (инициализации и копирования), деструктор, метод *read* ввода коэффициентов многочлена с клавиатуры, метод перевода многочлена в строковую форму, метод нахождения производной от многочлена. Перегрузите операции “=” присваивания, “+” и “\*” сложения и умножения двух многочленов, “[ ]” индексации (обращение к коэффициенту многочлена) и “()” вычисления значения многочлена в точке. Продемонстрируйте работу всех реализованных методов для двух многочленов с коэффициентами типа `int` и двух многочленов с коэффициентами типа `double`.
282. Создайте шаблон класса для работы с матрицами. Параметр-тип шаблона должен определять тип данных для элементов матрицы, параметры-константы – размерность матрицы. Реализуйте 2 конструктора (инициализации и копирования), деструктор, метод *read* ввода матрицы с клавиатуры, метод *print* вывода матрицы на экран и метод умножения матрицы на скаляр. Перегрузите операции “=” присваивания, “==” сравнения двух матриц, “+” и “\*” сложения и умножения двух матриц. Продемонстрируйте работу всех реализованных методов для двух матриц размера 3x3 и для двух матриц размера размера 3x2 и 2x3 с элементами типа `float` и `unsigned int` соответственно.
283. Создайте шаблон класса для моделирования стека с помощью массива. Параметр-тип шаблона должен определять тип данных для элементов стека, параметр-константа – максимальную глубину стека. Реализуйте 2 конструктора (инициализации и копирования), деструктор, метод перевода содержимого стека в строковую форму, методы добавления и

удаления элемента из стека, метод нахождения числа элементов в стеке. Перегрузите операции “=” присваивания, “==” сравнения двух стеков. Продемонстрируйте работу всех реализованных методов для двух стеков максимальной глубины 10 с элементами типа double и для двух стеков максимальной длины 15 и 10 с элементами типа char. Используя методы класса, перепишите содержимое одного стека поверх вершины другого стека.

284. Создайте шаблон класса для моделирования очереди (типа FIFO) с помощью массива. Параметр-тип шаблона должен определять тип данных для элементов очереди, параметр-константа – максимальную длину очереди. Реализуйте 2 конструктора (инициализации и копирования), деструктор, метод перевода содержимого очереди в строковую форму, методы добавления и удаления элемента из очереди, метод нахождения числа элементов в очереди и метод нахождения первого элемента в очереди. Перегрузите операции “=” присваивания, “==” сравнения двух очередей. Продемонстрируйте работу всех реализованных методов для двух очередей максимальной длины 10 с элементами типа float и для двух очередей максимальной длины 15 и 10 с элементами типа short. Используя методы класса, проверьте, являются ли элементы обеих очередей упорядоченными по убыванию. В случае положительного ответа создайте третью очередь, куда перепишите (не используя вспомогательных массивов) содержимое двух очередей, не нарушая упорядоченности. В противном случае создайте третью очередь, куда перепишите сначала элементы первой очереди, а затем элементы второй.

285. Даны  $n$  дат формата “Год, Месяц, Число”. Используя шаблон vector и структуру struct Date{unsigned int Year, Month, Day};, выведите на экран все даты в хронологическом порядке.

286. Даны  $n$  отметок времени формата “Часы:Минуты:Секунды”. Используя шаблон vector и структуру struct Time{unsigned int Hour, Min, Sec};, выведите на экран все отметки, исключая повторы.



287. Даны  $n$  слов, состоящих из букв английского алфавита. Используя шаблон `vector`, выведите на экран все слова в лексикографическом порядке.
288. Даны  $n$  комплексных чисел. Используя шаблон `vector` и структуру `struct Complex{float re, im};`, удалите все числа с наибольшим значением модуля.
289. Даны  $n$  точек на плоскости. Используя шаблон `vector` и структуру `struct Point{float x, y};`, выведите на экран сначала все точки из первой координатной четверти, затем из второй, третьей и четвертой.
290. Даны  $n$  дат формата “Год, Месяц, Число”. Используя шаблон `list` и структуру `struct Date{unsigned int Year, Month, Day};`, выведите на экран сначала все даты до 1 июня 1999 года, а затем – все даты после.
291. Даны  $n$  отметок времени формата “Часы:Минуты:Секунды”. Используя шаблон `list` и структуру `struct Time{unsigned int Hour, Min, Sec};`, выведите на экран все отметки, кроме самой ранней.
292. Даны  $n$  слов. Используя шаблон `list`, удалите все слова минимальной длины.
293. Дана последовательность из  $n$  комплексных чисел. Используя шаблон `list` и структуру `struct Complex{float re, im};`, вставьте после каждого третьего числа число с наименьшим значением модуля (если таких чисел несколько, взять первое из них).
294. Даны  $n$  точек на плоскости. Используя шаблон `list` и структуру `struct Point{float x, y};`, найдите сумму векторов с началом в самой удаленной от начала координат точке и концами в остальных точках.
295. Даны  $n$  дат формата “Год, Месяц, Число”. Используя шаблон `deque` и структуру `struct Date{unsigned int Year, Month, Day};`, выведите на экран сначала все даты из отрезка 10 апреля 2002 – 18 сентября 2003, а затем – все остальные.
296. Даны  $n$  отметок времени формата “Часы:Минуты:Секунды”. Используя шаблон `deque` и структуру `struct Time{unsigned int Hour, Min, Sec};`, выведите на экран все отметки, исключая наиболее часто встречаемую отметку.

297. Даны  $n$  слов. Используя шаблон `deque`, выведите на экран все слова в порядке уменьшения их длин.
298. Дана последовательность из  $n$  комплексных чисел. Используя шаблон `deque` и структуру `struct Complex{float re, im};`, выведите на экран числа с первого до последнего ненулевого в обратном порядке.
299. Даны  $n$  точек на плоскости. Используя шаблон `deque` и структуру `struct Point{float x, y};`, найдите любую пару наиболее удаленных друг от друга точек.

## 2.2. Динамические структуры данных

### 2.2.1. Массив с контролем индексов

Для решения задач 300-326 используйте массивы с контролем индексов, для чего необходимо создать класс для работы с одномерными массивами вещественных чисел, обеспечивающий контроль индексов. В классе обязательно должны быть определены конструкторы и деструктор. Кроме того, необходимо перегрузить операцию индексации “[ ]” таким образом, чтобы исключить возможность обращения к элементу массива, индекс которого выходит за пределы допустимых индексов данного массива.

300. Дан массив  $a(n)$ . Перепишите его в массив  $b$  таким образом, чтобы в массиве  $b$  были расположены сначала положительные, затем отрицательные, и затем нулевые элементы.
301. Дан массив  $a(n)$ . Перепишите его в массив  $b$  таким образом, чтобы в массиве  $b$  были расположены сначала элементы, стоящие в  $a$  на четных позициях, а затем элементы с нечетных позиций.
302. Дан массив  $a(n)$ . Постройте массив  $b(m)$ , в который перепишите все элементы из  $a$ , значения которых принадлежат отрезку  $[s;t]$ .

303. Перепишите положительные элементы массива  $a(n)$  в массив  $b$ . Отсортируйте массив  $b$  по убыванию.
304. Дан массив  $a(n)$ . Найдите среднее арифметическое элементов массива и элемент, наиболее отличный от среднего арифметического.
305. Дан массив  $a(n)$ . Вычислите произведение элементов массива, расположенных между его первым и последним единичными элементами. Если таких элементов нет, выведите соответствующее сообщение.
306. Дан массив  $a(n)$ . Вычислите сумму модулей элементов массива, расположенных после первого элемента, равного нулю. Если таких элементов нет, выведите соответствующее сообщение.
307. Дан массив  $a(n)$ . Перепишите в массив  $b$  те элементы из  $a$ , индексы которых являются числами Фибоначчи (1, 2, 3, 5, 8, 13, ...).
308. Дан массив  $a(n)$ . Проверьте, является ли  $a$  симметричным относительно своего центра.
309. Дан массив  $a(n)$ . Проверьте, образуют ли элементы массива знакопеременную последовательность.
310. Дан массив  $a(n)$ . Вычислите  $y = a_0 + a_0a_1 + a_0a_1a_2 + \dots + a_0a_1a_2\dots a_{m-1}$ , где  $m$  – либо номер первого неположительного элемента массива, либо число  $n$ , если в массиве все элементы положительны.
311. Дан массив  $a(n)$ . Найдите номер первого нулевого элемента массива и проверьте на убывание оставшуюся часть массива.
312. Дан массив  $a(n)$ . Для каждого числа, входящего в массив, укажите, сколько раз оно содержится в массиве. Результат запишите в новый массив.
313. Дан массив  $a(n)$ . Определите наибольшее количество одинаковых элементов в массиве.
314. Дан массив  $a(n)$ . Выясните, имеется ли в массиве хотя бы одна пара одинаковых чисел.
315. Дан массив  $a(n)$ . Выведите на экран все числа, входящие в массив по одному разу.
316. Дан массив  $a(n)$ . Найдите число различных элементов массива.

317. Даны массивы  $x(n)$ ,  $y(n)$ ,  $z(n)$  координат  $n$  точек в трехмерном пространстве. Определите, какие из точек наиболее и наименее удалены от точки  $M(p, q, r)$ .
318. Даны координаты точек на плоскости:  $x_1, y_1, \dots, x_n, y_n$ . Найдите номера двух точек, расстояние между которыми наибольшее.
319. Дан массив  $a(n)$ . Удалите из  $a$  все вхождения максимального и минимального элементов.
320. Дан массив  $a(n)$ . Без использования дополнительного массива необходимо “сжать” массив  $a$ , выбросив из него нулевые элементы. Если нулевых элементов нет, сообщите, что сжатие невозможно.
321. Дан массив  $a(n)$ . Элементы, расположенные в  $a$  между минимальным и максимальным элементами (включая эти элементы), расположите в обратном порядке.
322. Даны массивы  $a(n)$  и  $b(m)$ . Проверьте, упорядочены ли по убыванию  $a$  и  $b$ . Если оба массива упорядочены по убыванию, то постройте новый массив  $c(n+m)$  слиянием массивов  $a$  и  $b$  без нарушения упорядоченности. В противном случае в массиве  $c$  расположите сначала все элементы массива  $a$ , затем все элементы массива  $b$ .
323. Дан массив  $a(n)$ . Найдите его второй по величине элемент.
324. Дан массив  $a(n)$ . Найдите наибольшее количество одинаковых элементов массива, идущих подряд.
325. Дан массив  $a(n)$ . Найдите минимальный из положительных элементов массива с нечетными номерами (учтите случай, когда такие элементы отсутствуют).
326. Дан массив  $a(n)$ . Осуществите циклический сдвиг его элементов на  $m$  позиций вправо ( $0 < m < n$ ).

### 2.2.2. Список

327. Напишите программу ввода элементов списка, состоящего из целочисленных значений, с клавиатуры,

- вставки после каждого элемента списка значения ноль и вывода результирующего списка на экран.
328. Напишите программу ввода элементов списка, состоящего из целочисленных значений, с клавиатуры, после каждого нечетного значения вставки квадрата этого числа и вывода результирующего списка на экран.
329. Создайте список для хранения целочисленных элементов, значения которых введите с клавиатуры. Найдите среднее арифметическое элементов списка, делящихся на 3. Результат выведите на экран.
330. Создайте список для хранения целочисленных элементов, значения которых введите с клавиатуры. Удалите из списка все нечетные элементы. Полученный список выведите на экран.
331. Создайте список для хранения целочисленных элементов, значения которых введите с клавиатуры. Удалите из списка все нулевые элементы, затем после каждого элемента, равного пяти, добавьте случайное число из диапазона от 1 до 10. Полученный список выведите на экран.
332. Создайте список для хранения целочисленных элементов, значения которых введите с клавиатуры. Поменяйте местами минимальный и максимальный элементы списка. Полученный список выведите на экран.
333. Создайте список для хранения целочисленных элементов, значения которых введите с клавиатуры. Все нечетные элементы замените их квадратами, а после каждого четного числа вставьте значение ноль. Полученный список выведите на экран.
334. Создайте список для хранения символов, значения которых введите из файла. Напишите программу создания нового списка, который содержит только четные символы исходного списка. Результат запишите в другой файл.
335. Создайте список для хранения символов, значения которых введите из файла. Напишите программу создания нового списка, в котором символы идут в обратном порядке, по сравнению с исходным списком. Результат запишите в другой файл.

336. Создайте список для хранения символов, значения которых введите из файла. Напишите программу, удаляющую из списка все цифры. Результат запишите в другой файл.
337. Создайте список для хранения символов, значения которых введите из файла. Напишите программу, определяющую количество пробелов в списке. Результат запишите в другой файл.
338. Создайте список для хранения символов, значения которых введите из файла. Напишите программу создания нового списка путем удаления пробелов из исходного. Результат запишите в другой файл.
339. Создайте список для хранения символов, значения которых введите из файла. Напишите программу удаления из списка всех гласных букв английского алфавита. Результат запишите в другой файл.
340. Создайте список для хранения символов, значения которых введите из файла. Напишите программу создания нового списка, полученного из исходного посредством попарной перестановки символов. Результат запишите в другой файл.
341. Создайте список для хранения символов, значения которых введите из файла. Напишите программу, которая заменяет в списке каждую последовательность из двух и более пробелов на один пробел. Результат запишите в другой файл.
342. Создайте список для хранения символов, значения которых введите из файла. Напишите программу, которая меняет все цифры, встречающиеся в списке, на пробел. Результат запишите в другой файл.
343. Создайте список для хранения символов, значения которых введите из файла. Напишите программу создания нового списка, который состоит только из цифр, взятых из исходного списка. Результат запишите в другой файл.
344. Создайте список для хранения символов, значения которых введите из файла. Напишите программу, определяющую порядковый номер в списке первой встретившейся цифры. Результат запишите в другой файл.

345. Создайте список для хранения символов, значения которых введете из файла. Напишите программу, подсчитывающую количество заглавных букв русского алфавита в списке. Результат запишите в другой файл.
346. Создайте список для хранения символов, значения которых введете из файла. Напишите программу создания нового списка, который состоит из исходного с заменой всех прописных букв строчными. Результат запишите в другой файл.
347. Создайте список для хранения символов, значения которых введете из файла. Напишите программу перестановки всех пробелов в списке со следующими за ними символами. Результат запишите в другой файл.

### 2.2.3. Стек. Реализация с помощью списка и массива

348. Напишите программу вывода последовательности чисел в обратном порядке, при этом все отрицательные элементы замените на их квадраты.
349. Напишите программу следующего преобразования последовательности: сначала выводятся все отрицательные элементы в обратном порядке, а затем положительные в прямом порядке.
350. Напишите программу преобразования последовательности чисел, расположив сначала положительные элементы, а затем отрицательные с сохранением порядка следования, используя стек.
351. Напишите программу преобразования последовательности чисел, расположив сначала положительные элементы, а затем отрицательные. Порядок следования всех элементов обратный.
352. Запишите в стек первые  $n$  чисел Фибоначчи, затем выведите их на экран в обратном порядке ( $n$  задается пользователем).
353. Запишите в стек последовательность, в которой каждый следующий элемент равен сумме трех предыдущих (1, 1, 2 –

- первые члены последовательности), затем выведите ее на экран в обратном порядке.
354. В обратном порядке выведите арифметическую прогрессию (первый член и шаг задаются пользователем).
  355. В обратном порядке выведите геометрическую прогрессию (первый член и знаменатель задаются пользователем).
  356. В обратном порядке выведите  $n$  степеней числа  $a$ , воспользовавшись формулой  $a^n = a \cdot a^{n-1}$  ( $n, a$  задаются пользователем).
  357. В обратном порядке выведите  $n$  нечетных чисел, начиная с  $a$  ( $a$  и  $n$  задаются пользователем).
  358. Найдите среднее арифметическое элементов стека.
  359. Найдите сумму элементов стека.
  360. Напишите программу вывода на экран четных элементов последовательности в обратном порядке.
  361. Напишите программу вывода на экран в обратном порядке тех элементов последовательности, которые делятся на заданное пользователем число.

#### **2.2.4. Очередь. Реализация с помощью списка и массива**

362. Напишите программу удаления из заданной числовой (целочисленной) последовательности всех нулей.
363. Напишите программу выбора из заданной целочисленной последовательности элементов, кратных  $a$  ( $a$  задается пользователем). Все найденные элементы выведите на экран в порядке их следования.
364. Выведите на экран последовательность чисел, получающуюся из двух данных целочисленных последовательностей одинакового размера чередованием элементов (сохраняя порядок следования).
365. Выведите на экран только четные элементы заданной целочисленной последовательности в том же порядке.
366. Все отрицательные элементы заданной последовательности замените нулями.



367. Заданную последовательность целых чисел разбавьте нулями (между двумя рядом стоящими элементами вставьте ноль).
368. В том же порядке выведите на экран все элементы последовательности, которые больше  $a$  ( $a$  задается пользователем).
369. Выведите на экран последовательность, получающуюся из данной целочисленной последовательности заменой знаков у всех элементов.
370. Из данной целочисленной последовательности выберите все простые числа. Выведите их на экран в порядке их вхождения в последовательность.
371. Из данной целочисленной последовательности выберите возрастающую подпоследовательность (с сохранением порядка).
372. Все положительные элементы последовательности целых чисел прологарифмируйте, все остальные элементы оставьте неизменными, порядок следования необходимо сохранить.
373. Из последовательности целых чисел выберите те, которые являются решениями уравнения  $x^3 - 2x^2 - 5x + 6 = 0$ . Если таких чисел нет, выведите соответствующее сообщение.
374. Выведите на экран те элементы из данной последовательности целых чисел, из которых можно извлечь целый квадратный корень. Если таких чисел нет, выведите соответствующее сообщение.
375. Преобразуйте последовательность так, чтобы каждый ее элемент являлся суммой всех элементов на предыдущих позициях (включая текущую позицию) исходной последовательности целых чисел.

### 2.2.5. Дерево

376. Из введенной последовательности выберите только нечетные элементы, упорядочив их по возрастанию.
377. Упорядочите по возрастанию квадраты элементов некоторой введенной последовательности.

378. Выведите на экран сначала упорядоченные по возрастанию положительные элементы, а затем отрицательные (тоже упорядоченные по возрастанию).
379. Из некоторой введенной последовательности выберите положительные элементы и упорядочите их по убыванию.
380. Сначала упорядочите по возрастанию отрицательные элементы, а затем положительные элементы упорядочите по убыванию.
381. Упорядочите по возрастанию все элементы последовательности, которые больше  $a$  ( $a$  задается пользователем).
382. Посчитайте количество повторений каждого элемента из некоторой введенной последовательности. Элементы выведите в порядке возрастания (с указанием количества повторений).
383. Выведите все положительные элементы введенной последовательности в порядке убывания с указанием количества вхождений этого элемента.
384. Выведите на экран все элементы последовательности, упорядоченные по возрастанию, с указанием номера этого элемента в исходной последовательности.
385. Напишите функцию для нахождения уровня указанного узла дерева.
386. Напишите функцию для нахождения глубины данного дерева.
387. Напишите функцию для нахождения длины внутреннего пути по данному дереву (если нет повторения элементов).
388. Напишите функцию, печатающую дерево в обратном порядке.
389. Напишите функцию, осуществляющую симметричный обход дерева.

# Справочный материал

## Общая структура программы

Директивы препроцессора Объявление глобальных переменных  Тип_результата имя_функции_1(Список_параметров) { Операторы } ... Тип_результата имя_функции_n(Список_параметров) { Операторы }  Тип_результата main(Список_параметров) { Операторы }
---

Константы	Переменные
– это величины, которые <u>не изменяют своего значения</u> в процессе выполнения программы.  <b>const тип имя=значение;</b>	– именованный участок памяти, в котором хранится значение определенного типа, <u>которое можно изменить</u> во время выполнения программы.  <b>тип список_переменных;</b>

## Основные типы данных

Тип	Описание	Размер	Диапазон
<i>bool</i>	логический	1 байт	<i>false</i> (ложь) – 0; <i>true</i> (истина) – любое значение не равное нулю
<i>char</i>	символьный	1 байт	-128...127

<i>int</i>	целый	Зависит от реализации
<i>float</i>	с плавающей точкой	Зависит от реализации
<i>double</i>	двойной точности	Зависит от реализации

**Спецификаторы для составления составных типов данных**

Спецификатор	Описание
<i>short</i>	короткий
<i>long</i>	длинный
<i>signed</i>	знаковый
<i>unsigned</i>	беззнаковый

**Тип *void***

Множество значений пусто. Используется:

- для определения функций, не возвращающих значения,
- для указания пустого списка аргументов функции,
- базовый тип для указателей,
- в операции приведения типов.

**Виды переменных**

	Локальные переменные	Глобальные переменные	Формальные параметры функций
Описание	Объявляются внутри функции и доступны только в ней.	Описываются до всех функций и доступны из любого места программы.	Описываются в списке параметров функции.
Пример	<pre>int main() { float s; s=4.5; } // В функции main определена локальная вещественная переменная s.</pre>	<pre>float s; int main() { s=4.5; } // Определена глобальная переменная s типа float.</pre>	<pre>int f1(int x); // x – формальный параметр функции.</pre>

### Статические и автоматические классы памяти

Специ-фикатор	Название	Описание	Область видимости
<i>auto</i>	автоматический	К автоматическому классу памяти относятся локальные переменные. По умолчанию не инициализируются.	Видны только внутри функции или блока, в которых определены.
<i>static</i>	статический	К статическому классу памяти относятся глобальные переменные (в том числе переменные, объявленные внутри функции со спецификатором <i>static</i> ). По умолчанию инициализируются нулем или эквивалентным значением.	Доступны из любого места программы.

### Статическая и внешняя область действия

Внешняя	Статическая
Имеют переменные, относящиеся к статическому классу памяти, объявленные вне функций. Доступны в другом файле с использованием <i>extern</i> .	Имеют переменные, относящиеся к статическому классу памяти, область действия которых ограничена одним файлом. Объявляются с использованием <i>static</i> .
// Файл 1 <i>int a</i> ; // Файл 2 <i>extern int a</i> ; // доступ к <i>a</i> из файла 1	// Файл 1 <i>static int a</i> ; // Файл 2 <i>extern int a</i> ; // Ошибка! Доступ к <i>a</i> из файла 1 закрыт

### Операции языка C++ в порядке убывания приоритета

Операция	Описание
::	доступ к глобальной переменной
()	Вызов функции
[]	Индексация
. ->	Выбор элемента
sizeof	размер
!	логическое отрицание
~	поразрядное отрицание
- +	Унарные минус и плюс
++ --	Инкремент, декремент
& *	взятие адреса, снятие ссылки
(type)	преобразование типа
new delete	Выделение, освобождение памяти
. * ->*	Выбор поля через указатель
* / %	Умножение, деление, модуль(остаток)
+ -	Сложение, вычитание
<<	сдвиг влево, вывод
>>	сдвиг вправо, ввод
< >	Меньше, больше,
<= >=	меньше или равно, больше или равно
== !=	Равно, не равно
&	поразрядная конъюнкция (И)
^	поразрядное исключающее ИЛИ
	поразрядная дизъюнкция (ИЛИ)
&&	логическое И
	логическое ИЛИ
?:	условная операция
=	присваивание
*=	умножение с присваиванием
/=	деление с присваиванием
+=	сложение с присваиванием
-=	вычитание с присваиванием
%=	остаток от деления с присваиванием
<<=	сдвиг влево с присваиванием
>>=	сдвиг вправо с присваиванием

$\&=$ $ \equiv$ $\wedge=$	поразрядная конъюнкция с присваиванием поразрядная дизъюнкция с присваиванием поразрядное исключающее ИЛИ с присваиванием
,	последовательное вычисление

### Операции битовой арифметики $\&$ , $|$ , $\wedge$ , $\sim$

Операция	Описание
Арифметическое И ( $\&$ )	Оба операнда переводятся в двоичную систему, затем над ними происходит логическое поразрядное умножение операндов по следующим правилам: $1\&1=1$ , $1\&0=0$ , $0\&1=0$ , $0\&0=0$ .
Арифметическое ИЛИ ( $ $ )	Оба операнда переводятся в двоичную систему, после чего над ними происходит логическое поразрядное сложение операндов по следующим правилам: $1 1=1$ , $1 0=1$ , $0 1=1$ , $0 0=0$ .
Арифметическое исключающее ИЛИ ( $\wedge$ )	Оба операнда переводятся в двоичную систему, после чего над ними происходит логическая поразрядная операция $\wedge$ по следующим правилам: $1\wedge1=0$ , $1\wedge0=1$ , $0\wedge1=1$ , $0\wedge0=0$ .
Арифметическое отрицание ( $\sim$ )	Применение унарной операции $\sim$ вызывает побитную инверсию двоичного представления числа.

### Логические операции

A	B	!A	A&B	A B
0	0	1	0	0
0	1	1	0	1
1	0	0	0	1
1	1	0	1	1

### Условная операция условие ? выражение1 : выражение2;

Если условие истинно (не равно 0), то результатом будет выражение1, в противном случае выражение2.

### Операция присваивания

$a=b$ – присваивает $a$ значение $b$ .	
Множественное присваивание выполняется справа налево: $a=b=c=0$ ;	Сочетается с арифметическими или битовыми логическими операциями: $a$ операция= $b$ – сокращение от $a=a$ операция $b$

### Операции инкремента и декремента

<b><math>a</math> – любое целое или символьное выражение, которое может принимать значение</b>	
$a++$	Увеличивает $a$ на 1; результат выражения – значение $a$ до увеличения.
$++a$	Увеличивает $a$ на 1; результат выражения – значение $a$ после увеличения.
$a--$	Уменьшает $a$ на 1; результат выражения – значение $a$ до уменьшения.
$--a$	Уменьшает $a$ на 1; результат выражения – значение $a$ после уменьшения.
<b><math>p</math> – указатель на любое выражение, которое может принимать значение</b>	
$p++$	Увеличивает $p$ на $sizeof(*p)$ ; результат выражения – значение $p$ до увеличения.
$++p$	Увеличивает $p$ на $sizeof(*p)$ ; результат выражения – значение $p$ после увеличения.
$p--$	Уменьшает $p$ на $sizeof(*p)$ ; результат выражения – значение $p$ до уменьшения.
$--p$	Уменьшает $p$ на $sizeof(*p)$ ; результат выражения – значение $p$ после уменьшения.

### Операции распределения памяти

$new\ type$	Выделяет пространство для типа $type$ и возвращает адрес.
$new\ type[n]$	Выделяет пространство для массива из $n$ элементов типа $type$ и возвращает адрес.
$delete\ a$	Освобождает память, на которую указывает $a$ .
$delete[]\ a$	Освобождает память, занятую массивом $a$ .



<b>Операция преобразования типа</b>	<b>Операция определения размера</b>
Приведение выражения к другому типу данных: <b>(тип) выражение;</b>	Определение размера типа или объекта в байтах: <b>sizeof (тип) или sizeof выражение</b>

### Стандартные математические функции

Обозначение	Действие
abs(x)	Модуль целого числа
fabs(x)	Модуль вещественного числа
sin(x)	Функция синус
cos(x)	Функция косинус
tan(x)	Функция тангенс
asin(x)	Арксинус x в диапазоне [-1,1]
acos(x)	Арккосинус x в диапазоне [-1,1]
atan(x)	Арктангенс x в диапазоне -p/2 до p/2
exp(x)	Экспонента
log(x)	Натуральный логарифм (x>0)
log10(x)	Десятичный логарифм (x>0)
sqrt(x)	Корень квадратный (x≥0)
pow(x,y)	x в степени y
floor(x)	Наименьшее ближайшее целое
ceil(x)	Наибольшее ближайшее целое

### Строковые операции из библиотеки string.h

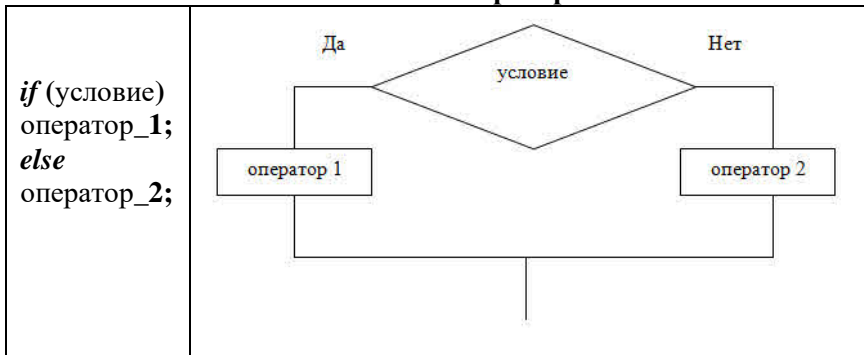
Прототип функции	Описание функции
size_t strlen( const char *s)	Вычисляет длину строки s в байтах (символ конца строки в результат не входит)
char *strcat( char *s1, const char *s2)	Присоединяет строку s1 в конец строки s2
char *strncat( char *s1,	Присоединяет maxlen символов строки s2 в конец строки s1

const char *s2, size_t maxlen)	
char *strcpy( char *s1, const char *s2)	Копирует строку s2 в s1
char *strncpy( char *s1, const char *s2, size_t maxlen)	Копирует maxlen символов строки s2 в место памяти, на которое указывает s1
char * strstr( char *s1, char *s2)	Возвращает подстроку строки s1, начиная с первого вхождения строки s2 в строку s1

### Функции преобразования строк из библиотеки `stdlib.h`

Прототип функции	Описание функции
double atof( const char *s)	Преобразует строку s в тип double
int atoi( const char *s)	Преобразует строку s в тип int
long atol( const char *s)	Преобразует строку s в тип long int
char *gcvt( double value, int digits, char *s)	Преобразует число типа double в строку s, digits – количество хранящихся значащих цифр.

### Условный оператор



## Оператор варианта

*switch* (выражение)

```

{
case значение_1: Операторы_1; break;
case значение_2: Операторы_2; break;
case значение_3: Операторы_3; break;
...
case значение_n: Операторы_n; break;
default: Операторы; break;
}
    
```



## Оператор цикла с предусловием

*while* (условие)

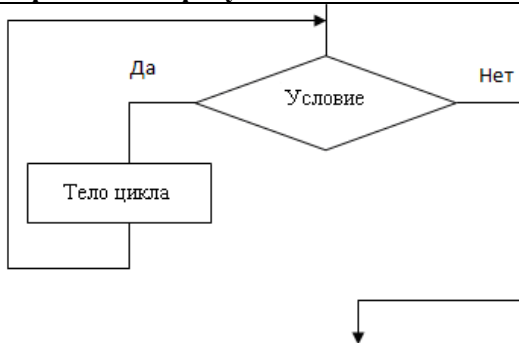
оператор;

или

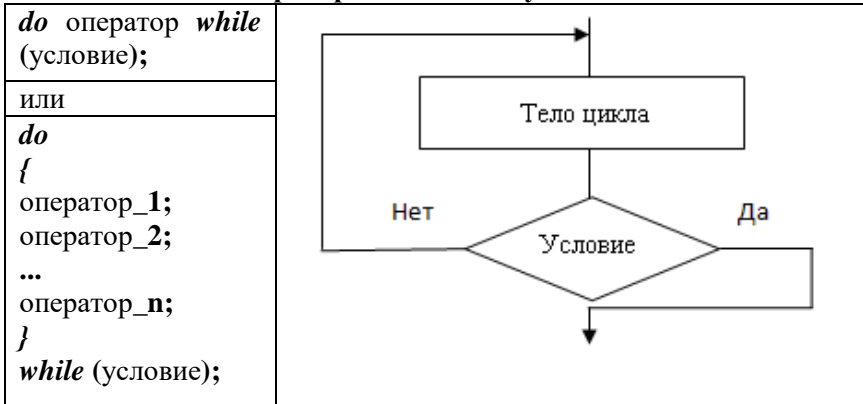
*while* (условие)

```

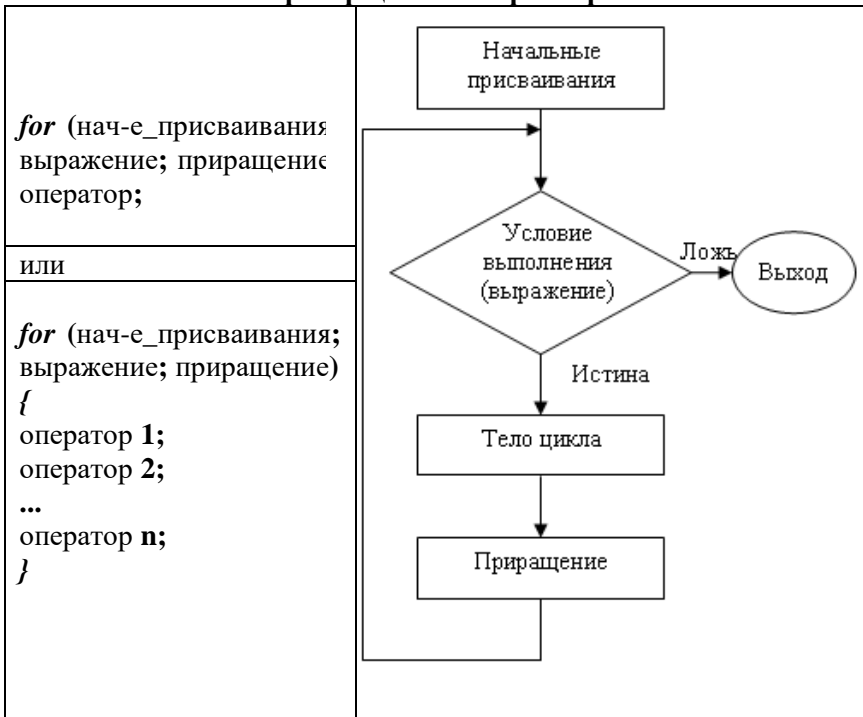
{
оператор 1;
оператор 2;
...
оператор n;
}
    
```



### Оператор цикла с постусловием



### Оператор цикла с параметром



### Операторы передачи управления

Оператор	Описание
<i>goto</i> метка;	Передаёт управление оператору с меткой: метка: оператор;
<i>break</i>	Немедленный выход из циклов <i>while</i> , <i>do-while</i> и <i>for</i> , а также из оператора выбора <i>switch</i> .
<i>continue</i>	Прерывает выполнение данного шага цикла.
<i>return</i>	Завершает выполнение функции и передаёт управление в точку ее вызова.

### Общие сведения о функциях

<b>Функция</b> – именованная, логически законченная группа операторов языка, которую можно вызвать для выполнения любое количество раз из различных мест программы.
<b>type</b> имя_функции ( <b>[type</b> параметр1, ..., <b>type</b> параметрN]) { тело_функции }
<b>Перегрузка</b> имени функции – создание двух или более функций с одним и тем же именем, отличающихся количеством параметров или типами параметров.
<b>Шаблон</b> функции – это особый вид функции, который использует различные типы исходных данных с одинаковым алгоритмом действия, описание шаблона начинается со служебного слова <code>template</code> , за которым в угловых скобках (<>) следует список используемых в функции типов данных: <code>template &lt; typename T1, ..., typename Tn &gt;</code>

### Указатели и ссылки

<b>Указатель</b> – это переменная, значением которой является адрес другой переменной или функции.	<b>Ссылка</b> – это второе имя для другой переменной.
<b>type * name;</b> name – переменная-указатель, указывает на переменную типа type.	<b>type &amp; name = name2;</b> name – переменная-ссылка, второе имя переменной name2 типа type.

## Массивы

<b>Массив</b> – структурированный тип данных, состоящий из фиксированного числа элементов одного типа.	
Статическое объявление	
Одномерный массив (вектор): <b>тип имя_массива[n];</b> <i>n</i> - количество элементов, нумерация: от 0 до <i>n</i> -1.	Двумерный массив (матрица): <b>тип имя_массива [n][m];</b> <i>n</i> – количество строк (0... <i>n</i> -1), <i>m</i> – кол-во столбцов (0... <i>m</i> -1).
В C++ можно описать многомерные массивы: <b>тип имя_массива [n1][n2]...[nk];</b>	
Динамическое объявление	
Одномерный массив (вектор): <b>тип* имя_массива =</b> <b>new тип[n];</b>	Двумерный массив (матрица): <b>тип* имя_массива =</b> <b>new тип[n*m];</b> или <b>тип** имя_массива;</b> <b>имя_массива =new тип*[n];</b> <b>for (int i=0;i&lt;n;i++)</b> <b>имя_массива[i]=new тип[m];</b>

## Структура

<b>Структура</b> – конструкция языка, предназначенная для логического и физического объединения данных произвольных типов.
<pre><i>struct</i> &lt;имя_структуры&gt; {     члены (элементы) структуры };</pre>

## Перечисление

<b>Перечисление</b> – это набор именованных целых констант.
<b>enum тег {список перечисления} список переменных;</b> Здесь <i>тег</i> и <i>список переменных</i> не являются обязательными, но что-то одно из них должно присутствовать.

## Объектно-ориентированное программирование. Классы

*Объектно-ориентированное программирование (ООП)* – это программирование, которое основывается на работе с объектами (а не с алгоритмами), каждый из которых является экземпляром определенного класса, а классы образуют иерархию наследования.

### Свойства ООП

*Инкапсуляция* – объединение данных и функций для их обработки в сочетании со скрытием детальной информации.

*Наследование* – свойство, позволяющее создавать иерархию классов, когда потомки наследуют все свойства своих предков, могут их изменять и добавлять новые.

*Полиморфизм* – возможность использования в различных классах иерархии одного имени для обозначения сходных по смыслу действий, выбирая подходящее действие во время выполнения программы.

### Класс

*Класс* – составной тип данных, определяемый пользователем, элементами которого являются переменные (поля данных) и функции (методы) для работы с ними.

*Экземпляр класса (или объект)* – конкретная величина типа данных «класс».

```
class name // name – имя класса
```

```
{  
    private:  
        // Описание закрытых членов класса  
    protected:  
        // Описание защищенных членов класса  
    public:  
        // Описание открытых членов класса  
};
```

public	открытые члены класса доступны любым функциям
private	доступ к закрытым членам открыт только

	методам класса и дружественным функциям класса
protected	для одиночных классов, не входящих в иерархию, равносильен private; доступ возможен членам класса и дружественным функциям произвольных классов

### Указатель this

*this* – скрытый параметр, в котором хранится указатель на вызвавший функцию объект.

```
class cl{
    int a;
    public:
    void method(){
        return a; // подразумевается this->a, т.е. поле
        // данных a вызвавшего функцию объекта
    }
    //...
};
int main(){
    cl object;
    cout<<object.method(); // метод method вызывает
    //объект object, поэтому this будет указывать на object
    return 0;
}
```

### Конструкторы

*Конструктор* – метод класса, предназначенный для инициализации объекта, запускается автоматически при создании экземпляра класса (при объявлении переменной типа class).

• Имя совпадает с именем класса.	• Не возвращает значение.
• Класс может иметь несколько конструкторов с разными параметрами.	• Конструктор по умолчанию – конструктор, вызываемый без параметров.



<ul style="list-style-type: none"> <li>• Можно задавать значения параметров по умолчанию. Их может содержать только один из конструкторов.</li> </ul>	<ul style="list-style-type: none"> <li>• Создается автоматически, если он не указан в классе (вызывает конструкторы по умолчанию для полей класса).</li> </ul>
<ul style="list-style-type: none"> <li>• Не наследуется.</li> </ul>	<ul style="list-style-type: none"> <li>• Нельзя описывать с модификаторами <b>const</b>, <b>virtual</b> и <b>static</b>.</li> </ul>
<p>Вызывается в случаях:</p> <p><b>имя_класса имя_объекта [(список параметров)];</b>  <b>имя_класса (список параметров);</b>  <b>имя_класса имя_объекта = выражение;</b></p>	

### Деструкторы

<p><b>Деструктор</b> – это метод класса, применяющийся для освобождения памяти, занимаемой объектом.</p>	
<ul style="list-style-type: none"> <li>• Имя деструктора состоит из тильды (~) и имени класса.</li> </ul>	<ul style="list-style-type: none"> <li>• Не имеет аргументов и возвращаемого значения.</li> </ul>
<p>Деструктор вызывается автоматически, когда объект выходит из области видимости:</p> <ul style="list-style-type: none"> <li>• для <i>локальных</i> объектов – при выходе из блока, в котором они объявлены;</li> <li>• для <i>глобальных</i> – как часть процедуры выхода из <b>main</b>;</li> <li>• для <i>объектов, заданных через указатели</i>, вызывается неявно при использовании операции <b>delete</b>.</li> </ul>	

### Статические элементы класса

<p><i>Статические поля</i> – это поля класса, объявленный с помощью модификатора <b>static</b>, которые существуют для всех объектов класса в единственном экземпляре, то есть не дублируются.</p>	
<ul style="list-style-type: none"> <li>• Память под статическое поле выделяется один раз при его инициализации.</li> </ul>	<ul style="list-style-type: none"> <li>• Статические поля доступны как через имя класса, так и через имя объекта.</li> </ul>
<ul style="list-style-type: none"> <li>• На статические поля распространяется действие спецификаторов доступа.</li> </ul>	<ul style="list-style-type: none"> <li>• Память, которую занимает статическое поле, не учитывается в операции <b>sizeof</b>.</li> </ul>

<p><i>Статические методы</i> – это методы, предназначенные для обращения к статическим полям класса, объявляются с помощью модификатора <b>static</b>.</p>	
<ul style="list-style-type: none"> <li>• Им не передается скрытый указатель <b>this</b>, поэтому могут обращаться только к статическим элементам класса.</li> </ul>	<ul style="list-style-type: none"> <li>• Обращение возможно либо через имя класса, либо через имя объекта.</li> </ul>

### Перегрузка операций

<p><i>Перегрузка операций</i> – расширение возможностей стандартных операций языка путем реализации способа их применения над типами классов.</p>	
<p><b>тип operator операция (список параметров) { тело функции }</b></p>	
<ul style="list-style-type: none"> <li>• Допустима перегрузка следующих операций: +, -, *, /, %, =, &lt;, &gt;, +=, -=, *=, /=, &lt;, &gt;, &amp;&amp;,   , ++, --, (), [], new, delete.</li> </ul>	<ul style="list-style-type: none"> <li>• Запрещено перегружать: ::, sizeof, ?:, .., *, #, ##.</li> </ul>
<ul style="list-style-type: none"> <li>• Нельзя изменить приоритет операций, ассоциативность и число операндов.</li> </ul>	<ul style="list-style-type: none"> <li>• Можно определить тремя способами: <ul style="list-style-type: none"> <li>- нестатический метод класса;</li> <li>- дружественная функция класса;</li> <li>- обычная функция.</li> </ul> </li> </ul>

### Дружественные функции и классы

<p><i>Дружественная функция</i> – функция, имеющая доступ к скрытым полям класса.</p>	
<p>Объявляется дружественной с помощью ключевого слова <b>friend</b>.</p>	
<ul style="list-style-type: none"> <li>• Объявляется <i>внутри</i> класса, к элементам которого ей нужен доступ.</li> </ul>	<ul style="list-style-type: none"> <li>• Указатель <b>this</b> ей не передается.</li> </ul>
<ul style="list-style-type: none"> <li>• Может быть обычной функцией или методом другого ранее определенного класса.</li> </ul>	<ul style="list-style-type: none"> <li>• На нее не распространяется действие спецификаторов доступа.</li> </ul>

<i>Дружественный класс</i> – класс, все методы которого имеют доступ к скрытым полям другого класса. Объявляется дружественным с помощью ключевого слова <b>friend</b> .	
<ul style="list-style-type: none"> <li>• Друзья друзей класса не являются друзьями данного класса.</li> </ul>	<ul style="list-style-type: none"> <li>• Не наследуются производными классами.</li> </ul>

### Наследование

<i>Наследование</i> – это способ повторного использования программного обеспечения, при котором новые производные классы (наследники) создаются на базе уже существующих базовых классов (родителей).		
<b>class имя :</b> <b>[private   protected   public] базовый_класс_1,</b> <b>***</b> <b>[private   protected   public] базовый_класс_n,</b> <b>{тело класса};</b>		
Модификатор наследования	Спецификатор базового класса	Доступ в производном классе
private	private protected public	нет private private
protected	private protected public	нет protected protected
public	private protected public	нет protected public

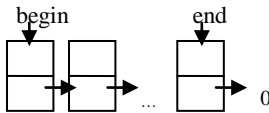
## Динамические структуры данных

*Динамическая структура данных* – такой способ организации данных, при котором память выделяется по мере необходимости отдельными блоками, связанными друг с другом с помощью указателей, размер таких данных изменяется во время выполнения программы.

### Списки

*Линейный список* – последовательность элементов, связанных между собой ссылками.

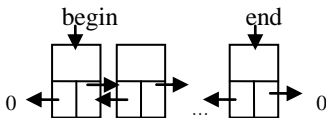
*Однонаправленный (односвязный) линейный список* – множество элементов, в котором каждый элемент содержит ссылку на следующий.



Описание элемента:

```
struct Node{
    int info;
    Node *next; //
    указатель на следующий
    элемент
};
```

*Двунаправленный (двусвязный) линейный список* – множество элементов, в котором каждый элемент содержит ссылку на предыдущий и на следующий элементы.



Описание элемента:

```
struct Node{
    int info;
    Node *next; //
    указатель на следующий
    элемент
    Node *prev; //
    указатель на предыдущий
    элемент
};
```

*Кольцевой список* – список, у которого последний элемент связан указателем с первым.

	<p>Описание элемента:</p> <pre> struct Node{     int info;     Node *next; //     указатель на следующий     элемент }; </pre>
--	--

### Операции над списком

– Добавление элемента в конец списка;
– чтение элемента с заданным ключом;
– вставка элемента в заданное место списка (до или после элемента с заданным ключом);
– удаление элемента с заданным ключом;
– упорядочивание списка по ключу и др.

### Удаление элемента списка

- 1 – выделение памяти под указатель на удаляемый элемент pkey
- 2 – проверка местоположения удаляемого элемента в списке

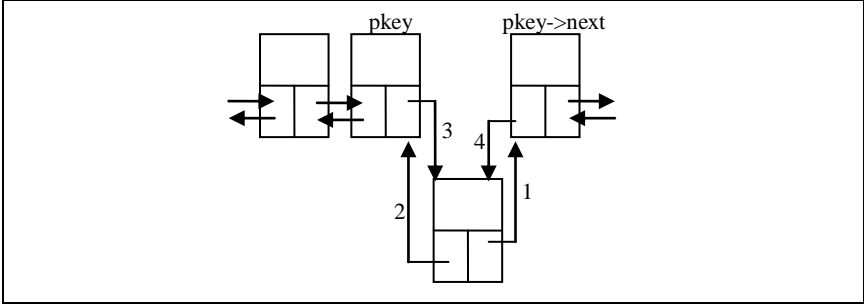
<p>В начале</p>	<p>3 – устанавливаем указатель на начало списка на следующий за удаляемым элемент</p> <p>4 – у нового начального элемента списка в поле указателя на предыдущий элемент устанавливаем значение ноль</p>	
-----------------	---	--

<p>В середине</p>	<p>3 – у предыдущего элемента устанавливаем указатель на следующий элемент, который следует за удаляемым 4 – у следующего за удаляемым элемента устанавливаем на предыдущий элемент, который предшествует удаляемому</p>	
<p>В конце</p>	<p>3 – устанавливаем указатель конца списка на предыдущий элемент 4 – для нового последнего элемента указатель на следующий элемент устанавливаем равным нулю</p>	

5 – освобождаем память из-под удаляемого элемента

### Вставка элемента в список

- 1 – установление связи нового узла с последующим
- 2 – установление связи нового узла с предыдущим
- 3 – установление связи предыдущего узла с новым
- 4 – установление связи последующего узла с новым



### Стек, очередь

<p><i>Стек</i> – это линейный список с одной точкой доступа, называемой вершиной. Данные могут быть добавлены только в вершину и удалены только из вершины стека.</p>	<p><i>Очередь</i> – это линейный список с двумя точками доступа: началом и концом. Данные могут быть добавлены только в конец очереди и удалены из ее начала.</p>
<p style="text-align: center;"><b>LIFO</b></p> <p>last in – first out (последний вошел – первый вышел)</p>	<p style="text-align: center;"><b>FIFO</b></p> <p>first in – first out (первый вошел – первый вышел)</p>
<p style="text-align: center;">Операции:</p> <ul style="list-style-type: none"> <li>– добавление элемента в вершину стека;</li> <li>– выборка элемента из вершины стека.</li> </ul>	<p style="text-align: center;">Операции:</p> <ul style="list-style-type: none"> <li>– добавление элемента в конец очереди;</li> <li>– выборка элемента из начала очереди.</li> </ul>

## Бинарное дерево

<p><i>Бинарное дерево</i> – динамическая структура данных, состоящая из узлов, каждый из которых содержит поле данных и два указателя на левое и правое поддерева.</p>	
	<p>Начальная точка бинарного дерева называется <i>корневым узлом</i>.</p>
	<p><i>Уровень</i> корня дерева равен 0. Уровень любого другого узла всегда больше на единицу, чем уровень узла, ссылающегося на данный.</p>
	<p><i>Глубиной</i> дерева называется максимальный из уровней узлов.</p>
	<p><i>Длина внутреннего пути</i> по дереву есть общее число ребер, находящихся в дереве.</p>
<p style="text-align: center;">Операции:</p> <ul style="list-style-type: none"> <li>– добавление элемента;</li> <li>– удаление элемента;</li> <li>– обход дерева;</li> <li>– упорядочивание элементов дерева и др.</li> </ul>	

## Обходы бинарных деревьев

Обход в прямом порядке	Симметричный обход	Обход в обратном порядке
<p>Каждый узел посещается до того, как посещены его потомки.</p>	<p>Посещаем сначала левое поддерево, затем узел, затем правое поддерево.</p>	<p>Узлы посещаются «снизу вверх».</p>
<ul style="list-style-type: none"> <li>• Посетить узел</li> <li>• Обойти левое поддерево</li> <li>• Обойти правое поддерево</li> </ul>	<ul style="list-style-type: none"> <li>• Обойти левое поддерево</li> <li>• Посетить узел</li> <li>• Обойти правое поддерево</li> </ul>	<ul style="list-style-type: none"> <li>• Обойти левое поддерево</li> <li>• Обойти правое поддерево</li> <li>• Посетить узел</li> </ul>



## Список литературы

1. *Абрамов С. А.* Задачи по программированию / С. А. Абрамов, Г. Г. Гнездилова, Е. Н. Капустина, М. И. Селюн. – М. : Наука, 1998. – 227 с.
2. *Астахова И. Ф.* Язык С++ : учебное пособие. – М. : Новое знание, 2003. – 208 с.
3. *Вирт Н.* Алгоритмы и структуры данных / Пер. с англ. Ткачев Ф.В. – М. : ДМК Пресс, 2010. – 272 с.
4. *Златопольский Д.М.* Сборник задач по программированию. – 3-е изд., перераб. и доп. – СПб. : БХВ-Петербург, 2011. – 304 с.
5. *Кнут Д. Э.* Искусство программирования, том 1. Основные алгоритмы, 3-е изд. – М. : Издательский дом Вильямс, 2001. – 720 с.
6. *Кнут Д. Э.* Искусство программирования, том 2. Получисленные алгоритмы, 3-е изд. – М. : Издательский дом Вильямс, 2001. – 832 с.
7. *Кнут Д. Э.* Искусство программирования, том 3. Сортировка и поиск, 2-е изд. – М. : Издательский дом Вильямс, 2001. – 824 с.
8. *Красиков И. В. Красикова И. Е.* Алгоритмы. Просто как дважды два. – М. : Эксмо, 2007. – 256 с.
9. *Лафоре Р.* Объектно-ориентированное программирование в С++. – 4-е изд. – М. : Питер, 2004. – 923 с.
10. *Липпман С. Б.* Язык программирования С++. Вводный курс / С. Б. Липпман, Ж. Лажойе, Б. Э. Му. – 4-е изд. : пер. с англ. – М. : ООО «И. Д. Вильямс», 2007. – 896 с.
11. *Папнас К., Мюррей У.* Программирование на С и С++. — Киев: «Ирина»; ВНУ, 2000. – 318 с.
12. *Подбельский В. В.* Язык С++ : учебное пособие. – 5-е изд. – М. : "Финансы и статистика", 2003. – 560 с.
13. *Романов Е.Л.* Практикум по программированию на С++: Уч. пособие. СПб. : БХВ-Петербург; Новосибирск: Изд-во НГТУ, 2004. – 432 с.
14. *Страуструп Б.* Язык программирования С++. – Киев : Диасофт, 2001. – 900 с.

15. *Шилдт Г.* Полный справочник по C++, 4-е изд. – М. : Издательский дом Вильямс, 2006. – 791 с.
16. *Шилдт Г.* Искусство программирования на C++. – СПб. : БХВ-Петербург, 2005. – 496 с.