

Учреждение образования
«БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ»

Н. Н. Пустовалова, Н. В. Пацей

ОСНОВЫ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ
в 2-х частях

Часть 1

Лабораторный практикум

Минск 2021

ПРЕДИСЛОВИЕ

Практикум содержит задания для выполнения лабораторных работ по дисциплине «Основы алгоритмизации и программирования». Основным предметом изучения является язык программирования C++, программы на котором выполняются на базе приложения **Microsoft Visual Studio**.

В каждой работе имеются краткие теоретические сведения по рассматриваемым вопросам. Преподаватель определяет, какие лабораторные работы должны выполнять студенты и в каком объеме. Предполагается, что выполнение большинства лабораторных работ занимает у студентов два академических часа.

При выполнении программ на компьютере возможны ситуации, когда возникают ошибки из-за того, что исходные данные заданы некорректно (деление на ноль, корень из отрицательного числа и т. п.). В таком случае надо ознакомиться с появившимися сообщениями об ошибках, проанализировать программу и выполнить с другими исходными данными.

В тексте практикума имеются кнопки, при нажатии на которые открываются тесты, предназначенные для контроля знаний студентов. Тестирование происходит по команде преподавателя и занимает несколько минут. Для работы тестирующих программ предварительно в приложении Word надо разрешить использование макросов. При этом тексты ответов на формах располагаются каждый раз случайным образом, и ответить на вопросы можно только один раз, так как после нажатия на кнопку «Результаты» форма с вопросами и вариантами ответов исчезает.

В приложении даны стандарты оформления кодов программ на языке C/C++.

Для **оформления отчетов по лабораторным работам** следует использовать приложение **Word**. Каждая работа должна содержать название работы, условия задач в соответствии со своим вариантом, алгоритмы, тексты разработанных программ, результаты. В верхнем колонтитуле записывается фамилия студента и номер группы, в нижнем – номера страниц.

Все работы сохраняются в **одном** файле.

ОГЛАВЛЕНИЕ

- Лабораторная работа № 1. [Системы счисления](#)
Лабораторная работа № 2. [Способы представления алгоритмов](#)
Лабораторная работа № 3. [Основные элементы языка C++. MS Visual Studio 2010](#)
Лабораторная работа № 4. [Ввод и вывод информации](#)
Лабораторная работа № 5. [Разветвляющиеся программы](#)
Лабораторная работа № 6. [Циклические программы](#)
Лабораторная работа № 7. [Отладка программ](#)
Лабораторная работа № 8. [Вычисление сумм, произведений, экстремумов](#)
Лабораторная работа № 9. [Решение инженерных задач на основе циклических программ](#)
Лабораторная работа № 10. [Одномерные массивы](#)
Лабораторная работа № 11. [Битовые операции](#)
Лабораторная работа № 12. [Использование указателей](#)
Лабораторная работа № 13. [Обработка символьной информации](#)
Лабораторная работа № 14. [Многомерные массивы](#)
Лабораторная работа № 15. [Динамическое выделение памяти](#)
Лабораторная работа № 16. [Функции пользователя](#)
Лабораторная работа № 17. [Указатели и ссылки при работе с функциями](#)
Приложение. [Стандарты оформления кода программы на языке C/C++](#)

Лабораторная работа № 1. Системы счисления

При организации вычислительных процессов в компьютерах используются десятичная система счисления (с/с), двоичная, восьмеричная и шестнадцатеричная.

Задание	Краткие теоретические сведения			
<p>1. Перевести несколько чисел (например, 12, 77, 436 и др.) из восьмеричной системы счисления в двоичную.</p> <p>Перевести несколько чисел (например, В8, 359, АА, 81 и др.) из шестнадцатеричной системы счисления в двоичную.</p>				
	Десятичная с/с	Двоичная с/с	Восьмеричная с/с	Шестнадцатеричная с/с
	0	0	0	0
	1	1	1	1
	2	10	2	2
	3	11	3	3
	4	100	4	4
	5	101	5	5
	6	110	6	6
	7	111	7	7
	8	1000	10	8
	9	1001	11	9
	10	1010	12	А
	11	1011	13	В
	12	1100	14	С
	13	1101	15	Д
	14	1110	16	Е
	15	1111	17	F
	16	10000	20	10
	17	10001	21	11
<p>Для перевода числа из 8-й и 16-й в 2-ю с/с надо каждую цифру числа заменить эквивалентной ей двоичной триадой (из таблицы слева) или тетрадой (четверкой цифр).</p> <p>Пример: $502_{(8\text{ с/с})} = \underline{101}\ \underline{000}\ \underline{010}_{(2\text{ с/с})}$</p>				

<p>2. Перевести несколько чисел (например, 10111001, 0110, 011 и др.) из двоичной системы счисления в восьмеричную.</p> <p>Перевести несколько чисел (например, 1111, 10101010 и др.) из двоичной системы счисления в шестнадцатеричную.</p>	<p>Для перевода числа из 2-й с/с в 8-ю и 16-ю нужно разбить целую часть числа влево от последнего разряда или от запятой (дробную часть числа – вправо от запятой) на триады или тетрады, и каждую такую группу заменить соответствующей восьмеричной или шестнадцатеричной цифрой. В случае необходимости неполные триады дополняются нулями.</p> <p>Пример: $1\ 111\ 110(2\ с/с) = 001\ 111\ 110(2\ с/с) = 176(8\ с/с)$ $0011\ 1101\ 0101, 1100(2\ с/с) = 3D5,C(16\ с/с)$</p>
<p>3. Перевести несколько чисел (например, 153, 236 и др.) из десятичной системы счисления в двоичную.</p>	<div data-bbox="705 635 952 892"> </div> <p>Чтобы перевести число из десятичной системы счисления в двоичную, надо разделить его на основание новой с/с, зафиксировать остаток от деления и частное. Затем частное снова разделить на основание с/с и зафиксировать остаток от деления. Процесс деления частных продолжается до тех пор, пока частное не станет меньше основания с/с.</p> <p>Все полученные в процессе деления остатки от деления и последнее частное будут образовывать цифры нужного результата в обратном порядке.</p> <p>Например, $25_{(10\ с/с)} = 11001_{(2\ с/с)} = 1 \cdot 2^4 + 1 \cdot 2^3 + 0 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0 = 25_{(10\ с/с)}$.</p>

4. Выполнить задания из таблицы, представленной ниже, в соответствии с вариантом. Номер варианта определяет преподаватель.

№ варианта	Условие
1	<p>Выполнить перевод чисел</p> $122_{(8\text{ c/c})} \rightarrow ?_{(2\text{ c/c})} \rightarrow ?_{(16\text{ c/c})};$ $110101_{(2\text{ c/c})} \rightarrow ?_{(10\text{ c/c})};$ $65_{(10\text{ c/c})} \rightarrow ?_{(2\text{ c/c})}$ $32_{(8\text{ c/c})} \rightarrow ?_{(2\text{ c/c})} \rightarrow ?_{(16\text{ c/c})};$ $10101_{(2\text{ c/c})} \rightarrow ?_{(8\text{ c/c})} \rightarrow ?_{(16\text{ c/c})};$ $86_{(10\text{ c/c})} \rightarrow ?_{(2\text{ c/c})}$
2	<p>Выполнить перевод чисел</p> $212_{(8\text{ c/c})} \rightarrow ?_{(2\text{ c/c})} \rightarrow ?_{(16\text{ c/c})};$ $110011_{(2\text{ c/c})} \rightarrow ?_{(8\text{ c/c})} \rightarrow ?_{(16\text{ c/c})};$ $322_{(10\text{ c/c})} \rightarrow ?_{(2\text{ c/c})}$ $55_{(8\text{ c/c})} \rightarrow ?_{(2\text{ c/c})} \rightarrow ?_{(16\text{ c/c})};$ $10101011_{(2\text{ c/c})} \rightarrow ?_{(10\text{ c/c})};$ $41_{(10\text{ c/c})} \rightarrow ?_{(2\text{ c/c})}$
3	<p>Выполнить перевод чисел</p> $25_{(8\text{ c/c})} \rightarrow ?_{(2\text{ c/c})} \rightarrow ?_{(16\text{ c/c})};$ $111111_{(2\text{ c/c})} \rightarrow ?_{(10\text{ c/c})};$ $122_{(10\text{ c/c})} \rightarrow ?_{(2\text{ c/c})}$ $134_{(8\text{ c/c})} \rightarrow ?_{(2\text{ c/c})} \rightarrow ?_{(16\text{ c/c})};$ $11000001_{(2\text{ c/c})} \rightarrow ?_{(8\text{ c/c})} \rightarrow ?_{(16\text{ c/c})};$ $68_{(10\text{ c/c})} \rightarrow ?_{(2\text{ c/c})}$
4	<p>Выполнить перевод чисел</p> $332_{(8\text{ c/c})} \rightarrow ?_{(2\text{ c/c})} \rightarrow ?_{(16\text{ c/c})};$ $1011_{(2\text{ c/c})} \rightarrow ?_{(10\text{ c/c})};$ $11111_{(10\text{ c/c})} \rightarrow ?_{(2\text{ c/c})}$ $12_{(8\text{ c/c})} \rightarrow ?_{(2\text{ c/c})} \rightarrow ?_{(16\text{ c/c})};$ $1111001_{(2\text{ c/c})} \rightarrow ?_{(8\text{ c/c})} \rightarrow ?_{(16\text{ c/c})};$ $73_{(10\text{ c/c})} \rightarrow ?_{(2\text{ c/c})}$
5	<p>Выполнить перевод чисел</p> $11_{(8\text{ c/c})} \rightarrow ?_{(2\text{ c/c})} \rightarrow ?_{(16\text{ c/c})};$ $1111_{(2\text{ c/c})} \rightarrow ?_{(10\text{ c/c})};$ $168_{(10\text{ c/c})} \rightarrow ?_{(2\text{ c/c})}$ $103_{(8\text{ c/c})} \rightarrow ?_{(2\text{ c/c})} \rightarrow ?_{(16\text{ c/c})};$ $1110111_{(2\text{ c/c})} \rightarrow ?_{(8\text{ c/c})} \rightarrow ?_{(16\text{ c/c})};$ $24_{(10\text{ c/c})} \rightarrow ?_{(2\text{ c/c})}$
6	<p>Выполнить перевод чисел</p> $1005_{(8\text{ c/c})} \rightarrow ?_{(2\text{ c/c})} \rightarrow ?_{(16\text{ c/c})};$ $10101_{(2\text{ c/c})} \rightarrow ?_{(10\text{ c/c})};$ $152_{(10\text{ c/c})} \rightarrow ?_{(2\text{ c/c})}$ $45_{(8\text{ c/c})} \rightarrow ?_{(2\text{ c/c})} \rightarrow ?_{(16\text{ c/c})};$ $11001_{(2\text{ c/c})} \rightarrow ?_{(8\text{ c/c})} \rightarrow ?_{(16\text{ c/c})};$ $56_{(10\text{ c/c})} \rightarrow ?_{(2\text{ c/c})}$

№ варианта	Условие
7	<p>Выполнить перевод чисел</p> $45_{(8 \text{ c/c})} \rightarrow ?_{(2 \text{ c/c})} \rightarrow ?_{(16 \text{ c/c})};$ $11101_{(2 \text{ c/c})} \rightarrow ?_{(10 \text{ c/c})};$ $43_{(10 \text{ c/c})} \rightarrow ?_{(2 \text{ c/c})}$ $651_{(8 \text{ c/c})} \rightarrow ?_{(2 \text{ c/c})} \rightarrow ?_{(16 \text{ c/c})};$ $1101001_{(2 \text{ c/c})} \rightarrow ?_{(8 \text{ c/c})} \rightarrow ?_{(16 \text{ c/c})};$ $104_{(10 \text{ c/c})} \rightarrow ?_{(2 \text{ c/c})}$
8	<p>Выполнить перевод чисел</p> $106_{(8 \text{ c/c})} \rightarrow ?_{(2 \text{ c/c})} \rightarrow ?_{(16 \text{ c/c})};$ $10011001_{(2 \text{ c/c})} \rightarrow ?_{(10 \text{ c/c})}$ $24_{(10 \text{ c/c})} \rightarrow ?_{(2 \text{ c/c})}$ $416_{(8 \text{ c/c})} \rightarrow ?_{(2 \text{ c/c})} \rightarrow ?_{(16 \text{ c/c})};$ $10100001_{(2 \text{ c/c})} \rightarrow ?_{(8 \text{ c/c})} \rightarrow ?_{(16 \text{ c/c})};$ $74_{(10 \text{ c/c})} \rightarrow ?_{(2 \text{ c/c})}$
9	<p>Выполнить перевод чисел</p> $31_{(8 \text{ c/c})} \rightarrow ?_{(2 \text{ c/c})} \rightarrow ?_{(16 \text{ c/c})};$ $1001_{(2 \text{ c/c})} \rightarrow ?_{(10 \text{ c/c})};$ $76_{(10 \text{ c/c})} \rightarrow ?_{(2 \text{ c/c})}$ $121_{(8 \text{ c/c})} \rightarrow ?_{(2 \text{ c/c})} \rightarrow ?_{(16 \text{ c/c})};$ $1110101_{(2 \text{ c/c})} \rightarrow ?_{(8 \text{ c/c})} \rightarrow ?_{(16 \text{ c/c})};$ $85_{(10 \text{ c/c})} \rightarrow ?_{(2 \text{ c/c})}$
10	<p>Выполнить перевод чисел</p> $54_{(8 \text{ c/c})} \rightarrow ?_{(2 \text{ c/c})} \rightarrow ?_{(16 \text{ c/c})};$ $10111011_{(2 \text{ c/c})} \rightarrow ?_{(10 \text{ c/c})};$ $40_{(10 \text{ c/c})} \rightarrow ?_{(2 \text{ c/c})}$ $214_{(8 \text{ c/c})} \rightarrow ?_{(2 \text{ c/c})} \rightarrow ?_{(16 \text{ c/c})};$ $11011_{(2 \text{ c/c})} \rightarrow ?_{(10 \text{ c/c})};$ $522_{(10 \text{ c/c})} \rightarrow ?_{(2 \text{ c/c})}$
11	<p>Выполнить перевод чисел</p> $131_{(8 \text{ c/c})} \rightarrow ?_{(2 \text{ c/c})} \rightarrow ?_{(16 \text{ c/c})};$ $10000001_{(2 \text{ c/c})} \rightarrow ?_{(10 \text{ c/c})};$ $98_{(10 \text{ c/c})} \rightarrow ?_{(2 \text{ c/c})}$ $24_{(8 \text{ c/c})} \rightarrow ?_{(2 \text{ c/c})} \rightarrow ?_{(16 \text{ c/c})};$ $1101111_{(2 \text{ c/c})} \rightarrow ?_{(8 \text{ c/c})} \rightarrow ?_{(16 \text{ c/c})};$ $21_{(10 \text{ c/c})} \rightarrow ?_{(2 \text{ c/c})}$
12	<p>Выполнить перевод чисел</p> $11_{(8 \text{ c/c})} \rightarrow ?_{(2 \text{ c/c})} \rightarrow ?_{(16 \text{ c/c})};$ $11111001_{(2 \text{ c/c})} \rightarrow ?_{(10 \text{ c/c})};$ $77_{(10 \text{ c/c})} \rightarrow ?_{(2 \text{ c/c})}$ $342_{(8 \text{ c/c})} \rightarrow ?_{(2 \text{ c/c})} \rightarrow ?_{(16 \text{ c/c})};$ $11011_{(2 \text{ c/c})} \rightarrow ?_{(8 \text{ c/c})} \rightarrow ?_{(16 \text{ c/c})};$ $1011_{(10 \text{ c/c})} \rightarrow ?_{(2 \text{ c/c})}$
13	<p>Выполнить перевод чисел</p> $1003_{(8 \text{ c/c})} \rightarrow ?_{(2 \text{ c/c})} \rightarrow ?_{(16 \text{ c/c})};$ $1010111_{(2 \text{ c/c})} \rightarrow ?_{(10 \text{ c/c})};$ $29_{(10 \text{ c/c})} \rightarrow ?_{(2 \text{ c/c})}$ $14_{(8 \text{ c/c})} \rightarrow ?_{(2 \text{ c/c})} \rightarrow ?_{(16 \text{ c/c})};$ $101111_{(2 \text{ c/c})} \rightarrow ?_{(8 \text{ c/c})} \rightarrow ?_{(16 \text{ c/c})};$ $468_{(10 \text{ c/c})} \rightarrow ?_{(2 \text{ c/c})}$

№ варианта	Условие
14	<p>Выполнить перевод чисел</p> $41_{(8 \text{ с/с})} \rightarrow ?_{(2 \text{ с/с})} \rightarrow ?_{(16 \text{ с/с})};$ $1001_{(2 \text{ с/с})} \rightarrow ?_{(10 \text{ с/с})};$ $66_{(10 \text{ с/с})} \rightarrow ?_{(2 \text{ с/с})}$ $1035_{(8 \text{ с/с})} \rightarrow ?_{(2 \text{ с/с})} \rightarrow ?_{(16 \text{ с/с})};$ $110101_{(2 \text{ с/с})} \rightarrow ?_{(8 \text{ с/с})} \rightarrow ?_{(16 \text{ с/с})};$ $852_{(10 \text{ с/с})} \rightarrow ?_{(2 \text{ с/с})}$
15	<p>Выполнить перевод чисел</p> $316_{(8 \text{ с/с})} \rightarrow ?_{(2 \text{ с/с})} \rightarrow ?_{(16 \text{ с/с})};$ $11100001_{(2 \text{ с/с})} \rightarrow ?_{(10 \text{ с/с})};$ $764_{(10 \text{ с/с})} \rightarrow ?_{(2 \text{ с/с})}$ $46_{(8 \text{ с/с})} \rightarrow ?_{(2 \text{ с/с})} \rightarrow ?_{(16 \text{ с/с})};$ $1011101_{(2 \text{ с/с})} \rightarrow ?_{(8 \text{ с/с})} \rightarrow ?_{(16 \text{ с/с})};$ $33_{(10 \text{ с/с})} \rightarrow ?_{(2 \text{ с/с})}$
16	<p>Выполнить перевод чисел</p> $653_{(8 \text{ с/с})} \rightarrow ?_{(2 \text{ с/с})} \rightarrow ?_{(16 \text{ с/с})};$ $111001_{(2 \text{ с/с})} \rightarrow ?_{(10 \text{ с/с})};$ $100_{(10 \text{ с/с})} \rightarrow ?_{(2 \text{ с/с})}$ $107_{(8 \text{ с/с})} \rightarrow ?_{(2 \text{ с/с})} \rightarrow ?_{(16 \text{ с/с})};$ $101001_{(2 \text{ с/с})} \rightarrow ?_{(8 \text{ с/с})} \rightarrow ?_{(16 \text{ с/с})};$ $28_{(10 \text{ с/с})} \rightarrow ?_{(2 \text{ с/с})}$

Тест "Системы счисления"

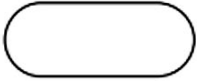


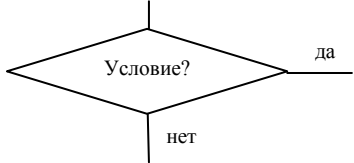




[В начало практикума](#)

Лабораторная работа № 2. Способы представления алгоритмов

Алгоритм – это система правил, определяющая последовательность действий над некоторыми объектами и приводящая к достижению поставленной цели после конечного числа шагов.

Для записи алгоритма решения задачи применяется:

- словесно-формульное описание;
- блок-схема (отдельные блоки представлены ниже).

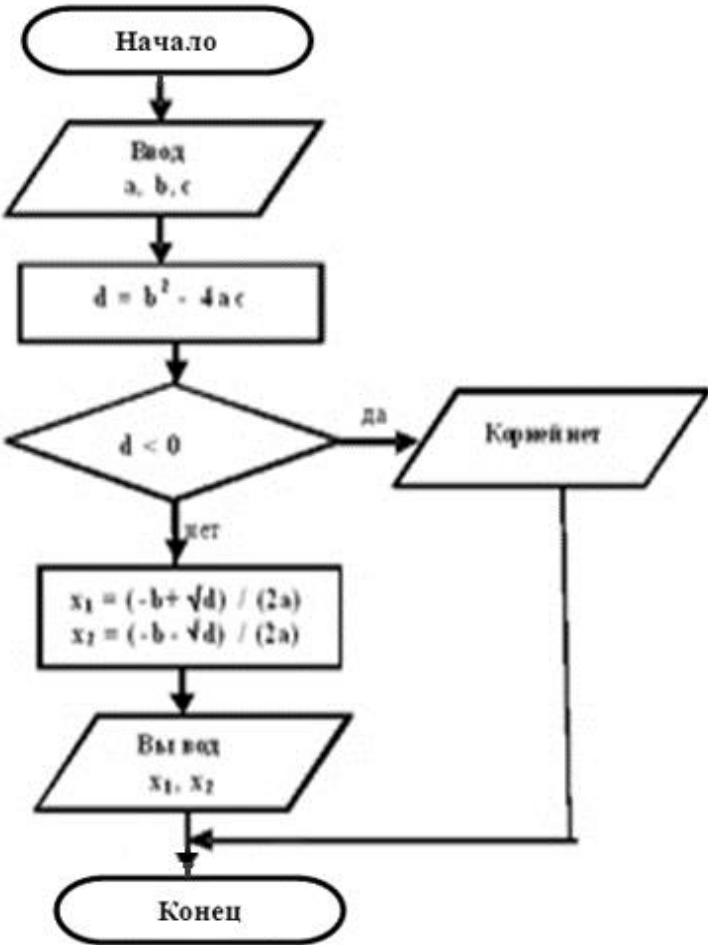
 <p>Блок начала-конца алгоритма</p>	 <p>Блок ввода-вывода данных</p>	 <p>Блок вычислений</p>	 <p>Условный блок</p>
 <p>Предопределенный процесс (использование готовых программ)</p>	 <p>Счетный цикл</p>	 <p>Комментарий</p>	 <p>Соединитель (используется при разрыве схемы)</p>

Различают алгоритмы линейные, разветвляющиеся, циклические.

Линейный – это такой алгоритм, в котором все действия выполняются последовательно друг за другом и только один раз. Схема представляет собой последовательность блоков, которые располагаются сверху вниз в порядке их выполнения.

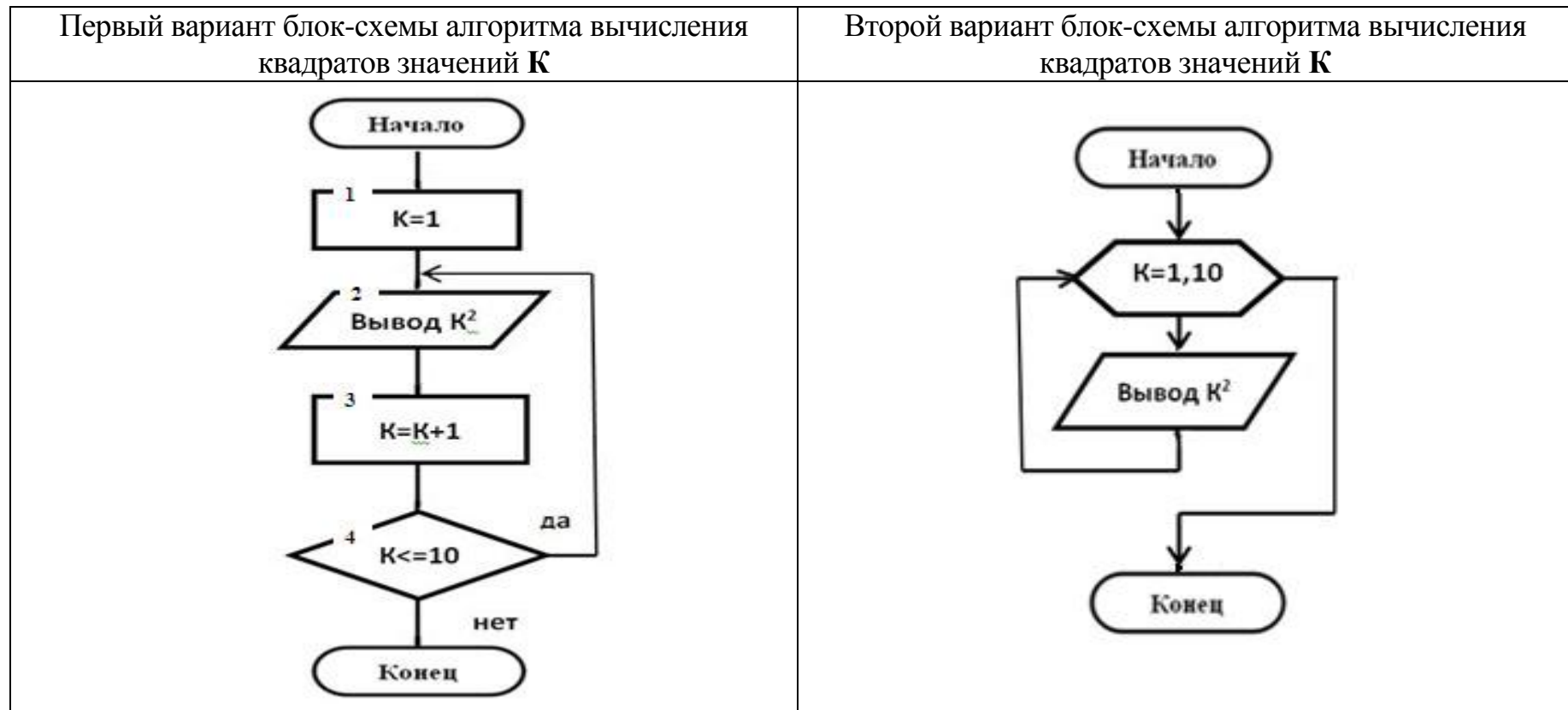
С помощью алгоритмов *разветвляющейся* структуры можно описать задачи, в которых выбор направления вычислений осуществляется по итогам проверки некоторого условия.

Для решения многих задач характерно многократное повторение отдельных участков вычислений. Для решения таких задач применяются алгоритмы *циклической* структуры.

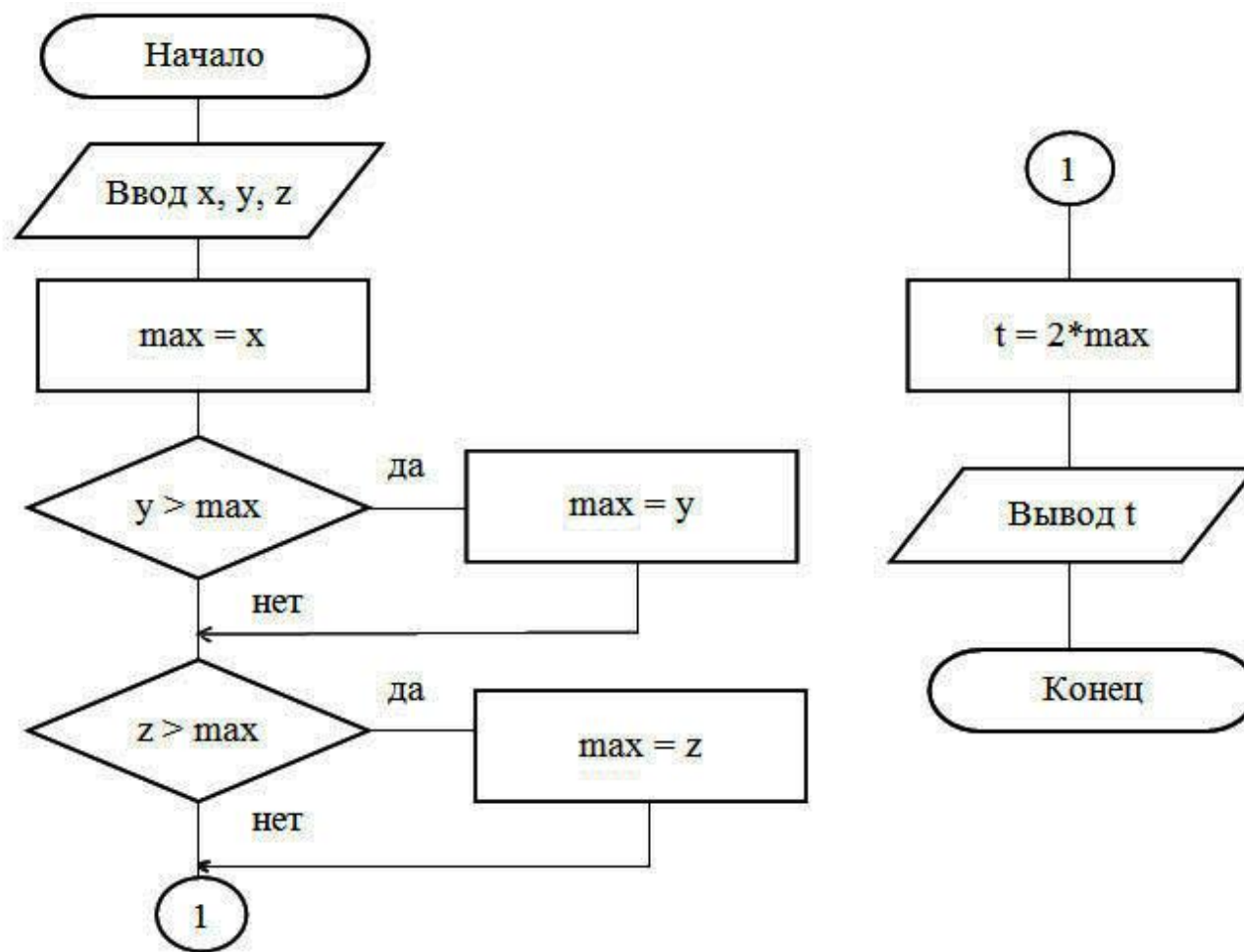
Словесно-формульное описание алгоритма решения квадратного уравнения	Блок-схема алгоритма решения квадратного уравнения
<p>1. Ввести числовые значения переменных a, b и c.</p> <p>2. Вычислить d по формуле $d = b^2 - 4ac.$</p> <p>3. Если d < 0, то перейти к п. 4, иначе перейти к п. 5.</p> <p>4. Напечатать сообщение «Корней нет» и перейти к п. 7</p> <p>5. Вычислить:</p> $x_1 = \frac{-b + \sqrt{d}}{2a}, \quad x_2 = \frac{-b - \sqrt{d}}{2a}$ <p>6. Напечатать значения x₁ и x₂</p> <p>7. Конец вычислений.</p>	 <pre> graph TD Start([Начало]) --> Input[/Ввод a, b, c/] Input --> CalcD[d = b² - 4ac] CalcD --> Decision{d < 0} Decision -- да --> NoRoot[/Корней нет/] Decision -- нет --> CalcX["x₁ = (-b + √d) / (2a) x₂ = (-b - √d) / (2a)"] CalcX --> Output[/Вывод x₁, x₂/] NoRoot --> End([Конец]) Output --> End </pre>

Пример. Словесно-формульное описание алгоритма вычисления квадратов значений K , изменяющихся от 1 до 10.

1. Определить значение $K = 1$.
2. Вычислить K^2 и вывести.
3. Увеличить значение K на единицу.
4. Если $K \leq 10$, то перейти к п. 2. Иначе перейти к п. 5.
5. Конец вычислений.



Все блоки схемы должны иметь одинаковую ширину. Вниз и вправо стрелки можно не ставить. При необходимости в схеме можно использовать нумерацию блоков в разрыве верхней линии слева и соединители.



Задание

1. В приложении Word разработать словесно-формульное описание и *блок-схему* алгоритма в соответствии со своим вариантом для решения задачи из таблицы. Номер варианта определяет преподаватель.

№ вариан- та	Условие задачи
1	Даны действительные числа x, y, z . Поменять значения в переменных так, чтобы $x \geq y \geq z$.
2	Есть действительные числа $x1, y1, x2$. Вычислить $\max(x1, y1, x2)$.
3	Даны действительные числа x, y, z . Вычислить $\max(x + y + z, xyz) * 3$.
4	Определить, есть ли среди заданных целых чисел A, B, C два чётных числа.
5	Даны действительные числа x, y, z . Поменять значения в переменных так, чтобы $x \leq y \leq z$.
6	Среди заданных целых чисел k, f, t найти пары кратных.
7	Даны числа a, b, c . Если $a < b < c$, то каждое число заменить наибольшим, если $a > b > c$, то числа оставить без изменений, в противном случае все числа заменить их квадратами.
8	Есть действительные числа a, b, k . Поменять значения переменных так, чтобы $a > b > k$.
9	Даны действительные числа $x, y, x2$. Вычислить $\max(x, y, x2) * \min(x, y, x2)$.
10	Определить, есть ли среди заданных целых чисел n, m, k нечётные числа.
11	Даны действительные числа x, y, z . Вычислить $\max(x, y, z) * \min(xy, yz, zx)$.
12	Есть действительные числа x, y, z . Вычислить $\min(x+y+z / 2, xyz) + \max(xy, y)$.

№ вариан- та	Условие задачи
13	Даны действительные числа x_1, y_1, y_2 . Вычислить $\max(x_1+y_1, x_1y_1, y_1 y_2)+\min(x_1, y_1, y_2)$.
14	Существуют действительные числа y_1, x_2, y_2 . Вычислить $\min(y_1+x_2y_2, y_1+x_2+y_2, y_2)+5$.
15	Даны действительные числа x, y, z . Вычислить $\min(x + y + z, xyz) * \min(x, y, z)$.
16	Определить, есть ли среди заданных целых чисел A, B, C хотя бы одно нечётное.

2. В соответствии со своим вариантом из таблицы лабораторной работы № 5 разработать словесно-формульное описание и *блок-схему* алгоритма для решения задачи.

Тест "Способы представления алгоритмов"

[В начало практикума](#)

Лабораторная работа № 3. Основные элементы языка C++.

Целое число на языке C++ может быть десятичным числом (**-15, 9...**), восьмеричным (**020, 0360**) или шестнадцатеричным (**0x2B, 0x10**). Содержится в переменной типа **int**, занимает в памяти **4** байта.

Вещественное число с плавающей точкой – это десятичное число с точкой или экспонентой (**115.75, 1.5E-2, -0.025**). Содержится в переменной типа **double** (занимает обычно **8** байтов) или в переменной типа **float** (**4** байта).

Символьное данное состоит из одного символа, заключенного в апострофы (' ', 'Q'). Содержится в переменной типа **char**, занимает **1** байт.

Логическое данное может принимать только значения **false** (соответствует 0) и **true** (1). Содержится в переменной типа **bool**, занимает **1** байт.

Если требуется, чтобы значение переменной не изменялось, то для ее описания используется *ключевое* слово **const**.

В выражениях применяются *знаки операций*: +, −, *, /.

Математические функции вычисляются с помощью стандартных встроенных функций.

Математическая запись	sinx	cosx	e^x	 x 	tgx	lnx	√x	x^y
Запись на C++	sin(x)	cos(x)	exp(x)	abs(x)	tan(x)	log(x)	sqrt(x)	pow(x, y)

Программа начинается с заголовка, который содержит описание стандартных файлов библиотек, пользовательских файлов и др. Заголовочные файлы включаются в программу с помощью директивы препроцессора **#include**. После заголовка следует оператор начала текста программы, например, **void main()**. В фигурных скобках {...} записывается тело функции.

Все элементы данных должны быть определены в программе перед их использованием. Определения данных и операторы всегда завершаются точкой с запятой. Один оператор может распространяться на несколько строк либо несколько операторов могут находиться в одной строке.

Программы на языке C++ выполняются в приложении **Microsoft Visual Studio**, для запуска которого надо выполнить: **Пуск / Программы / Microsoft Visual Studio**.

Задание	Краткие теоретические сведения
<p>1. Создать новый <i>проект</i> с именем, которое состоит из фамилии студента и номера лабораторной работы.</p> <p>В проекте создать новый <i>файл</i> с программой под именем main.cpp</p>	<p>Для создания нового <i>проекта</i> надо в окне Начальная страница выполнить Создать проект. Среда Visual Studio отобразит окно Создать проект. В окне следует развернуть узел Visual C++, выбрать Win32 и на центральной панели выбрать Консольное приложение Win32. Затем в поле Имя надо ввести имя проекта, в поле Расположение указать путь размещения проекта, или выбрать его с помощью кнопки Обзор.</p> <p>После нажатия ОК откроется окно Мастер приложений Win32. Надо нажать кнопку Далее, в появившемся окне поставить галочку в поле Пустой проект. После нажатия кнопки Готово на экране появится окно Обозреватель решений.</p> <p>В контекстном меню окна Обозреватель решений выполнить Добавить / Создать элемент. В левой части появившегося окна выбрать Код, в центре выбрать Файл C++. В поле Имя задать имя нового файла, указать расширение ".cpp", нажать Добавить.</p>
<p>2. Ввести программу, текст которой приведен справа, проанализировать ее.</p> <p>Выполнить программу, ознакомиться с результатом.</p>	<div data-bbox="629 743 1131 1125" data-label="Text"> <pre>[Введи/* Пример 1 */ #include <iostream> void main() { float x = 3, y; y = x * x + sin(x); std::cout << y; }</pre> </div> <p>Программа рассчитывает значение $y = x^2 + \sin(x)$ для $x = 3$.</p> <p>Она начинается с комментария, который заключен внутри комбинации символов /* и */.</p> <p>Строка #include <iostream> будет заменена содержимым файла <iostream.h>, который находится в стандартной библиотеке и отвечает за ввод-вывод данных на экран.</p> <p>В строке float x = 3, y; определяется тип переменных x и y как действительных чисел. В переменную x помещается число 3.</p> <p>С помощью оператора std::cout значение y выводится на экран.</p>

	<p>При наборе кода автоматически отображаются список методов, параметры методов и функций. При наведении мыши на элемент дается его краткое описание.</p> <p>При наборе текста программы одинаковые прописные и строчные буквы считаются <i>различными</i> символами.</p> <p>Если допущена ошибка в ключевом слове, то слово подчеркивается красной волнистой чертой.</p> <p>Если номера строк не отображаются, то надо выполнить Сервис / Параметры / Текстовый редактор / C++. Отметить пункт Показывать номера строк.</p> <p>Для компиляции программы следует выполнить Построение / Компилировать (или нажать клавиши Ctrl – F7).</p> <p>Запуск программы на выполнение: Отладка / Запуск без отладки (или Ctrl – F5).</p>		
<p>3. В предыдущем проекте заменить текст программы на программу, записанную в правой части данного пункта, выполнить ее.</p>	<p>Исходные данные: k = 4; a = 4.1; x = 5·10⁻⁵. Вычислить: t = 2tg(k)/a + ln(3 + x) + e^x, u = √(t + 1) – sin(x)cos(t)</p>	<pre>#include <iostream> void main() { double t, u, k = 4, a = 4.1, x = 5e-5; t = 2 * tan(k) / a + log(3 + x) + exp(x); u = sqrt(t + 1) - sin(x) * cos(t); std::cout << "t=" << t; std::cout << "u=" << u; }</pre>	

4. В таблице ниже приведены условия задач. В соответствии со своим вариантом разработать программу для решения задачи. Опробовать работу программы и проанализировать результаты.

Возможны ситуации, когда исходные данные заданы некорректно и при выполнении программы возникают ошибки (деление на ноль, корень из отрицательного числа и т. п.). В таком случае надо выполнить программу с другими исходными данными.

№	Формулы для вычислений	Исходные данные	№	Формулы для вычислений	Исходные данные
1	$t = 2k/a + \ln(2 + x)$ $u = \sqrt{k - 1}/(t + 1)$	$k = 4; x = 2 \cdot 10^{-4};$ $a = 8,1$	9	$z = e^{cx} / y + 3$ $u = \sqrt{z^3 - 0,1z}$	$y = 0,5; c = 1,4;$ $x = 2 \cdot 10^{-4}$
2	$z = 1/(x - 1) + \sin x - \sqrt{n}$ $y = (e^{-b} + 1)/2z$	$n = 2; b = -0,12;$ $x = 1,3 \cdot 10^{-4}$	10	$y = (z + \ln z)/(e^{-3} + \sqrt{a})$ $s = (1 + m \cdot n)/\ln(1 + z)$	$z = 1,7; a = 4 \cdot 10^{-8};$ $m = 3; n = 3$
3	$d = \operatorname{tg}(-x \cdot i) / \sqrt{x - z}$ $f = \sin(2d) / d$	$i = -6; x = 4,5;$ $z = 1,5 \cdot 10^{-6}$	11	$w = \operatorname{tg} 1 \cdot (1 + x) + z - e^a$ $d = 9\sqrt{2 - 3 \cdot x} + a + 1 $	$a = 1,5; x = -1,8$ $z = 15 \cdot 10^{-9}$
4	$p = 2,6t + \cos(y/(3x + y))$ $q = \sin t / \cos t$	$t = 6; y = -1,2;$ $x = 0,4 \cdot 10^6$	12	$t = \operatorname{tg} x + r \cdot (1 - \ln x)$ $u = t/(x^3 + 1)/(1 - e^{k-4})$	$k = 6; r = 5 \cdot 10^{-7}$ $x = 0,095$
5	$w = (a + b) \operatorname{tg} x / (x + 1)$ $v = 1/2 \cdot b - \sqrt{w - a \cdot b}$	$b = 40; x = 1,1;$ $a = 5 \cdot 10^{-6}$	13	$t = (d \cdot c + a \cdot \sqrt{c - 1})$ $y = 0,5t / d + e^a$	$d = 0,5 \cdot 10^{-8}; c = -9;$ $a = 1,5$

№	Формулы для вычислений	Исходные данные	№	Формулы для вычислений	Исходные данные
6	$t = 1 / \sqrt{y} + 14a$ $u = (t + 1) / (a + 2)$ $s = \ln((2n / 3) + e^{-n} / u)$	$y = 0,956;$ $a = 5 \cdot 10^{-6};$ $n = 4$	14	$s = \sqrt{t \cdot a / t + 1} + 4e^{2 \cdot b}$ $w = s \cdot a / (1 + 0,1a)$ $v = s + j \cdot \sqrt{a^2 + b^2}$	$a = 1,5; b = -8,1;$ $j = 4; t = 4 \cdot 10^{-8}$
7	$y = \sqrt{1 + x} - \cos(2 / m)$ $w = 0,6z - 2e^{-2 \cdot y \cdot m}$	$x = 1,4; m = 6;$ $z = 0,05 \cdot 10^{-5}$	15	$y = a \cdot e^{-2b} - \sqrt{1 + a}$ $r = \ln(1 + 20b) / (1 + a)$	$a = 1,75;$ $b = 4,5 \cdot 10^{-4}$
8	$s = 0,4x - 1 / j \cdot \operatorname{tg} y$ $t = s - \sin s$	$x = 0,1; j = 12;$ $y = 5 \cdot 10^{-6}$	16	$y = \cos(5m) / \sin^2(0,4m)$ $w = 4z \cdot y - 7e^{-2 \cdot y}$	$m = 6;$ $z = 0,05 \cdot 10^{-5}$

5. К номеру своего варианта прибавить 3 и написать программу для новых исходных данных (для вариантов с 14 по 16 перейти к вариантам с 1 по 3).

Тест "Основные элементы языка C++"

[В начало практикума](#)

Лабораторная работа № 4. Ввод и вывод информации

На языке C++ используются следующие способы ввода-вывода информации: потоковый, форматированный, символьный.

Задание	Краткие теоретические сведения	
<p>1. Выполнить программу, записанную справа, которая использует <i>потоковый</i> вывод данных. Проанализировать ее текст.</p> <p>Добавить в программу ввод и вывод переменных различных типов.</p>	<pre>void main() { setlocale(LC_CTYPE, "Russian"); int t; cout << "Введите t="; cin >> t; cout << "t=" << t << endl; cout << "Тип Размер в байтах" << endl; cout << "int: " << sizeof(int) << endl; cout << "char: " << sizeof(char) << endl; cout << "float: " << sizeof(float) << endl; cout << "double: " << sizeof(double)<< endl; // sizeof определяет размер объекта в байтах }</pre>	<p>В программе оператор setlocale (LC_CTYPE, "Russian"); используется для вывода русского текста.</p> <p>cout – стандартный поток вывода на экран дисплея; cin – стандартный поток ввода с клавиатуры</p> <p>endl – перевод на новую строку.</p> <p>Если в программе записано using namespace std; (используя стандартное пространство имен), то перед каждым оператором вывода можно не писать std::</p> <p>// – начало комментария, конец которого определяется концом строки.</p> <p>Выражение cin >> x; используется для ввода числа с клавиатуры в переменную x. В данном примере вводится целое число, которое при выполнении программы нужно набрать на клавиатуре и нажать клавишу Enter.</p>

2. Опробовать работу программы, приведенной в правой части.

Добавить в программу ввод и вывод переменных с использованием манипуляторов.

```
#include <iomanip>
void main()
{
    setlocale(LC_CTYPE, "Russian");
    using namespace std;
    char c, probel; probel = ' ';
    cout << "Введите символ "; cin >> c;
    cout << setw(35) << setfill(probel) << probel;
    cout << setw(10) << setfill(c) << c << endl;
    cout << setw(34) << setfill(probel) << probel;
    cout << setw(12) << setfill(c) << c << endl;
    cout << setw(33) << setfill(probel) << probel;
    cout << setw(14) << setfill(c) << c << endl;
}
```

setw(int n) – установить ширину поля вывода в **n** символов;

setfill(int n) – символ-заполнитель;

setprecision(int n) – установить количество цифр после запятой;

setbase(int n) – установить систему счисления для вывода чисел (**n** может принимать значения 0, 2, 8, 10, 16, причем 0 означает систему счисления по умолчанию, т.е. 10).

При использовании некоторых манипуляторов надо добавлять директиву **#include <iomanip>**.

Пример вывода числа из 6 символов (3 до запятой и 2 после):

using namespace std;

.....

double x;

cout << setw(6) << setprecision(2) << x << endl;

При выводе данных могут использоваться *манипуляторы*:

endl – перейти на новую строку;

ends – вывести нулевой байт (признак конца строки символов);

dec – вывод числа в десятичной системе;

oct – вывод числа в восьмеричной системе;

hex – вывод числа в шестнадцатеричной системе ;

<p>3. Выполнить программу, записанную в правой части, которая использует <i>форматированный</i> ввод-вывод данных.</p> <p>Изменить программу так, чтобы выводилась своя фамилия, имя и отчество, факультет, номер группы русскими буквами.</p> <p>При выводе использовать управляющие коды.</p> <p>Внести изменения в программы лабораторной работы № 3 с тем, чтобы осуществлялся форматированный ввод и вывод данных.</p>	<p>Можно управлять перемещением курсора на экране и выполнять некоторые другие функции, используя <i>управляющие коды</i>:</p> <p><code>\n</code> - перемещает курсор в начальную позицию следующей строки; <code>\t</code> – перемещает курсор в следующую позицию табуляции экрана; <code>\r</code> – перемещает курсор к началу той же строки без перехода на следующую; <code>\b</code> – передвигает курсор на одну позицию влево.</p> <p><u>Пример</u> программы, выводящей слово "Privet".</p> <table border="1" data-bbox="600 448 2092 796"> <tr> <td data-bbox="600 448 1227 796"> <pre>#include <stdio.h> #include <conio.h> void main() { printf("\n\t Privet\n"); printf("\n... Press key"); _getch(); }</pre> </td><td data-bbox="1227 448 2092 796"> <p>Заголовочный файл с именем stdio.h может использоваться при форматированном вводе-выводе данных.</p> <p>Заголовочный файл conio.h. поддерживает функцию _getch(), которая ожидает нажатия клавиши на клавиатуре.</p> <p>Функция printf() печатает информацию.</p> </td></tr> </table> <p>Каждому аргументу функции printf() соответствует своя <i>спецификация</i> преобразования, которая начинается с символа процента (%), после которого стоит буква, указывающая тип данного:</p> <p>%d – десятичное целое число; %f – вещественное число типа float или double; %c – символ; %s – строка; %p – указатель; %u – беззнаковое целое число; %o – целое число в восьмеричной системе счисления;</p>	<pre>#include <stdio.h> #include <conio.h> void main() { printf("\n\t Privet\n"); printf("\n... Press key"); _getch(); }</pre>	<p>Заголовочный файл с именем stdio.h может использоваться при форматированном вводе-выводе данных.</p> <p>Заголовочный файл conio.h. поддерживает функцию _getch(), которая ожидает нажатия клавиши на клавиатуре.</p> <p>Функция printf() печатает информацию.</p>
<pre>#include <stdio.h> #include <conio.h> void main() { printf("\n\t Privet\n"); printf("\n... Press key"); _getch(); }</pre>	<p>Заголовочный файл с именем stdio.h может использоваться при форматированном вводе-выводе данных.</p> <p>Заголовочный файл conio.h. поддерживает функцию _getch(), которая ожидает нажатия клавиши на клавиатуре.</p> <p>Функция printf() печатает информацию.</p>		

	<p>%x – целое число в шестнадцатеричной системе счисления;</p> <p>%e – вещественное число в экспоненциальной форме.</p> <p>После символа % можно указывать строку цифр, задающую минимальную ширину поля вывода, например, %5d (для целого числа), %4.2f (для вещественного числа – две цифры после запятой для поля шириной 4 символа). Если указанной ширины не хватает, происходит автоматическое расширение.</p> <p>Функция scanf() позволяет вводить данные любых типов. В качестве аргументов используются указатели объектов &. Например, scanf(“ %d%f ”, &x, &y);</p> <p>Если нужно ввести значение строковой переменной, то использовать символ & не нужно. Строка – массив символов, а имя массива эквивалентно адресу его первого элемента:</p> <pre>char name[20]; scanf(“%s”, name);</pre> <p>В новых версиях Visual Studio компилятор C++ считает функцию scanf небезопасной и выдаёт ошибку во время компиляции. Можно использовать безопасную версию scanf_s либо добавить строку для отключения этой ошибки:</p> <pre>#define _CRT_SECURE_NO_WARNINGS</pre>
<p>4. Изучить <i>символьный</i> ввод и вывод данных. Написать программу, реализующую диалог, используя пример, записанный в правой части.</p>	<div><pre>#include <windows.h> void main() { SetConsoleOutputCP(1251); SetConsoleCP(1251); char name[60]; puts("Как вас зовут? "); gets_s(name); printf("Привет, %s\n", name); }</pre></div> <p>Функция gets() (или gets_s) вводит строку в переменную, записанную в скобках.</p> <p>Функция puts() (или puts_s) осуществляет вывод символов на экран. Параметром puts() может быть строковая переменная:</p> <pre>char a[] = “Привет!”; puts (a);</pre>

	<p>Функция putchar() предназначена для вывода единичного символа на экран. Параметром функции может быть символ (например, putchar ('H');) или символьная переменная:</p> <pre>char letter; letter = 'G'; putchar (letter);</pre> <p>Функция getchar() вводит с клавиатуры единичный символ:</p> <pre>int letter; letter = getchar();</pre> <p>Для корректного ввода и вывода кириллицы на консоль в примере использованы функции: SetConsoleOutputCP() и SetConsoleCP(). В качестве параметра обеим функциям передается номер кодовой страницы кириллицы – это 1251.</p>
--	--

6. В соответствии со своим вариантом написать программы по условиям, приведенным в таблице ниже. Для ввода и вывода информации использовать *поток*овый и *форматированный* способы.

№ варианта	Условие задачи
1	1. Нарисовать полукруг, закрашенный некоторым введенным символом.
	2. Известна диагональ квадрата. Вычислить его площадь.
2	1. Нарисовать шестиугольник по центру консольного окна, закрашенный введенным символом.
	2. Разделить число m на две части, разность которых равна 5.

№ варианта	Условие задачи
3	1. Нарисовать елочку, закрашенную введенным символом.
	2. Поменять значения двух переменных: а) используя дополнительную переменную; б) не используя дополнительной переменной.
4	1. Нарисовать круг по центру консольного окна, закрашенный введенным символом.
	2. Даны числа X, Y, Z . Определить среднее арифметическое и среднее геометрическое их модулей.
5	1. Нарисовать треугольник по центру консольного окна, закрашенный введенным символом.
	2. Животновод в начале зимы повышает цену на молоко на p %, а летом снижает на столько же процентов. Изменится ли цена на молоко и если да, то в какую сторону и на сколько, через n лет?
6	1. Нарисовать звезду, закрашенную введенным символом.
	2. Определить расстояние на плоскости между двумя точками с заданными координатами M1(x1,y1) и M2(x2,y2) .
7	1. Нарисовать трапецию по центру консольного окна, закрашенную введенными символами.
	2. Известна длина окружности. Найти площадь круга, ограниченного этой окружностью.
8	1. Нарисовать человечка, закрашенного введенным символом.
	2. Вычислить площадь треугольника по формуле Герона, если заданы его стороны.
9	1. Нарисовать разнонаправленные стрелки ($\rightarrow \uparrow \downarrow \leftarrow$), состоящие из введенного символа.
	2. Дано действительное число a . Не пользуясь никакими другими операциями, кроме умножения, получить a⁴ за две операции.

№ варианта	Условие задачи
10	1. Нарисовать овал, закрашенный введенным символом.
	2. N школьников делят K яблок поровну, остаток остается в корзинке. Сколько яблок достанется каждому школьнику?
11	1. Нарисовать квадрат, закрашенный введенным символом.
	2. Найти площадь и основание такого прямоугольника, основание которого в 2 раза больше высоты, а площадь равна периметру.
12	1. Нарисовать знак +, закрашенный введенным символом.
	2. Сформировать число, представляющее собой реверсную (обратную в порядке следования разрядов) запись заданного трехзначного числа.
13	1. Нарисовать сердце ♥, закрашенное введенным символом.
	2. В бригаде, работающей на уборке сена, имеется N косилок. Первая из них работала M часов, а каждая следующая на 10 мин. Больше, чем предыдущая. Сколько часов работала вся бригада?
14	1. Нарисовать ромб ♦, закрашенный введенным символом.
	2. В компьютерной игре игрок выигрывает 50 очков, если он сбивает самолет; 100 очков, если он сбивает ракету; 200 очков, если он сбивает спутник. Определить число очков игрока, который сбил A самолетов, B ракет и C спутников.
15	1. Нарисовать домик, закрашенный введенным символом.
	2. Определить номера подъезда и этажа по номеру квартиры девятиэтажного дома с четырьмя подъездами, считая, что на каждом этаже ровно 4 квартиры.

№ варианта	Условие задачи
16	1. Нарисовать грибок, закрашенный введенным символом.
	2. Автомобиль на каждом из пяти одинаковых по длине участках дороги шел с известной средней скоростью. Разработать программу для определения средней скорости на всем пути.

Тест "Ввод и вывод информации"

[В начало практикума](#)

Лабораторная работа № 5. Разветвляющиеся программы

В разветвляющихся программах производится выбор направления вычислений в зависимости от выполнения некоторых условий.

Задание	Краткие теоретические сведения
<p>1. Выполнить программу, приведенную в правой части, несколько раз с различными значениями переменной j.</p> <p>Переделать программу с тем, чтобы ввод и вывод осуществлялся с помощью <i>поточковых</i> операторов ввода-вывода.</p>	<div><pre>#include <iostream> void main() { setlocale(LC_CTYPE, "Russian"); double b, z, x = 4.e-3, s = 1.1, j; cout<<"Введите j "; cin>>j; b = s + (5 * x + j); if(b < 1.5) z = sqrt(x + 0.3 * j) + b; else z = abs(x * j + b); cout<<"j="<<j<<" z="<< z<<endl; }</pre></div> <p>Если выражение b < 1,5 истинно, то z вычисляется по первой формуле, в противном случае – по второй.</p> <p>Вместо оператора if можно использовать <i>тернарный</i> оператор:</p> <p>z = b < 1.5 ? sqrt(x + 0.3 * j) + b: abs(x * j + b);</p> <p><u>Пример.</u> Вычислить значение z по формулам:</p> <p>b = s + (5*x + j),</p> $z = \begin{cases} \sqrt{x+0,3j} + b, & b < 1,5 \\ /xj + b/, & b \geq 1,5 \end{cases},$ <p>где x = 4·10⁻³; s = 1,1;</p> <p>j = {2; 6,8; 0,03; 55; ...}.</p> <p>Программа выполняется несколько раз с различными значениями переменной j.</p> <p>При каждом повторении программы в операторе if проверяется условие в скобках.</p>

<p>2. Выполнить программу, приведенную в правой части.</p> <p>Изменить программу с тем, чтобы ввод и вывод осуществлялся с помощью <i>поточковых</i> операторов ввода-вывода.</p>	<p><u>Пример.</u> Ввести с клавиатуры вещественные числа x, y, z, используя функции форматированного ввода. Вычислить max(x + y + z, xyz) · min(x, y, z).</p> <pre> #include <iostream> void main() { setlocale(LC_CTYPE, "Russian"); float x, y, z, mn, mx; printf("Введите x "); scanf_s("%f", &x); printf("Введите y "); scanf_s("%f", &y); printf("Введите z "); scanf_s("%f", &z); if ((x + y + z) < (x * y * z)) mx = (x * y * z); else mx = (x + y + z); mn = x; if (mn > y) mn = y; if (mn > z) mn = z; printf("max(x + y + z, xyz) * min(x, y, z) = %f \n", mn*mx); } </pre>
<p>3. Выполнить программу с использованием оператора выбора switch, приведенную в правой части,</p> <p>Записать условие задачи.</p>	<p>Для организации выбора из множества различных вариантов используется оператор выбора switch. Формат оператора:</p> <pre> switch (выражение) { [case константное выражение 1]: [список операторов 1]; [case константное выражение n]: [список операторов n]; [default: [список операторов];] } </pre>

```

#include <iostream>
int main()
{
    setlocale(LC_CTYPE, "Russian");
    int k;
    puts("Вы хотите купить автомобиль? (1-да, 2-нет)");
    std::cin >> k;
    switch (k)
    {
        case 1: { puts("Какой? (1- Майбах, 2-Тойота, 3-Лада)");
                std::cin >> k;
                switch (k)
                {
                    case 1: puts("Не слишком дорого?"); break;
                    case 2: puts("Хороший выбор"); break;
                    case 3: puts("Сомнительный выбор");
                        break;
                }
                break;
        }
        case 2: puts("Полезно ходить пешком"); break;
        default: puts("Некорректный вариант"); break;
    }
    return 0;
}

```

оператор, следующий после **switch**.

Оператор **break** передает управление оператору, следующему за **switch**.

Выражение в круглых скобках может быть *целой* или *символьной* константой.

Выполняется оператор **switch** следующим образом: вычисляется выражение в круглых скобках и полученное значение последовательно сравнивается с константными выражениями, следующими за ключевыми словами **case**.

Если одно из выражений совпадает со значением константного выражения, то управление передается оператору, записанному за соответствующим ключевым словом **case**.

Если совпадений нет, то управление передается на оператор, помеченный ключевым словом **default**, а в случае его отсутствия – на

	<p>Оператор return 0 завершает выполнение функции и возвращает системе значение 0.</p> <p>В некоторых случаях приходится использовать оператор безусловного перехода goto, который передает управление оператору, помеченному меткой (например, оператор goto a; передает управление оператору: a: std::cin>>k; Здесь a – метка оператора).</p> <p>Без крайней необходимости не рекомендуется использовать оператор goto.</p>
--	--

4. Написать и выполнить программу по первой блок-схеме лабораторной работы № 2.

5. В соответствии со своим вариантом написать программу по условию, приведенному в таблице ниже. Использовать операторы *if* и (или) *тернарные* операторы.

№ варианта	Условие задачи
1	Ввести три числа m, n, p . Подсчитать количество отрицательных чисел.
2	Определить, имеется ли среди трёх чисел a, b и c хотя бы одна пара равных между собой чисел.
3	Даны три числа a, b и c . Найти среднее геометрическое этих чисел, если все они отличны от нуля, и среднее арифметическое в противном случае.
4	По номеру месяца напечатать пору года.
5	Определить, есть ли среди заданных целых чисел A, B, C, D хотя бы одно нечётное.
6	Дано натуральное трехзначное число n . Верно ли, что среди его цифр есть 0 или 1?
7	В переменную Y ввести номер года. Определить, является ли год високосным.

№ варианта	Условие задачи
8	Дано натуральное четырехзначное число n . Верно ли, что все его цифры различны?
9	Проверить, является ли дробь A / B правильной.
10	Число делится на 3 тогда, когда сумма его цифр делится на 3. Проверить этот признак на примере заданного трехзначного числа.
11	Для целого числа K от 1 до 9 напечатать фразу «мне K лет», учитывая при этом, что при некоторых значениях K слово «лет» надо заменить на слово «год» или «года».
12	Есть натуральное двузначное число n . Верно ли, что среди его цифр есть 1 или 9?
13	Для натурального числа K напечатать фразу «мы нашли K грибов в лесу», согласовав окончание слова «гриб» с числом K .
14	Даны три числа a, b, c . Значение наибольшего из них присвоить переменной d .
15	Определить есть ли среди заданных целых чисел A, B, C, D хотя бы одно чётное.
16	По введенному числу (от 0 до 7) напечатать название цифры.

6. Написать программу, реализующую диалог на любую тему с использованием оператора **switch**.

7. Дополнительные задачи.

1. Введены с клавиатуры три числа. Найти сумму тех чисел, которые делятся на 5. Если таких чисел нет, то вывести текст «Error». Использовать только тернарные операторы.

2. На шахматном поле в клетке (**k, l**) расположен слон. Угрожает ли он фигуре стоящей в клетке (**m, n**)? Предусмотреть обработку ошибочного ввода пользователя (выдавать диагностическое сообщение).

3. Возможно ли на прямоугольном участке застройки размером **a** на **b** метров разместить два дома размером **p** на **q** и **r** на **s** метров? Дома можно располагать только параллельно сторонам участка.
4. Пройдет ли шар радиуса **r** через ромбообразное отверстие с диагоналями **p** и **q**?
5. Можно ли коробку размером **a** × **b** × **c** упаковать в посылку размером **r** × **s** × **t**? «Углом» укладывать нельзя.
6. На шахматной доске стоят черный король и белые ладья и слон. Проверить, есть ли угроза королю и если есть, то от кого именно.
7. Два отрезка на плоскости заданы координатами своих концов. Определить, имеют ли эти отрезки общие точки.
Замечание: рассмотреть различные случаи взаимной ориентации отрезков: на одной прямой, на параллельных или пересекающихся прямых.

Тест "Разветвляющиеся программы"

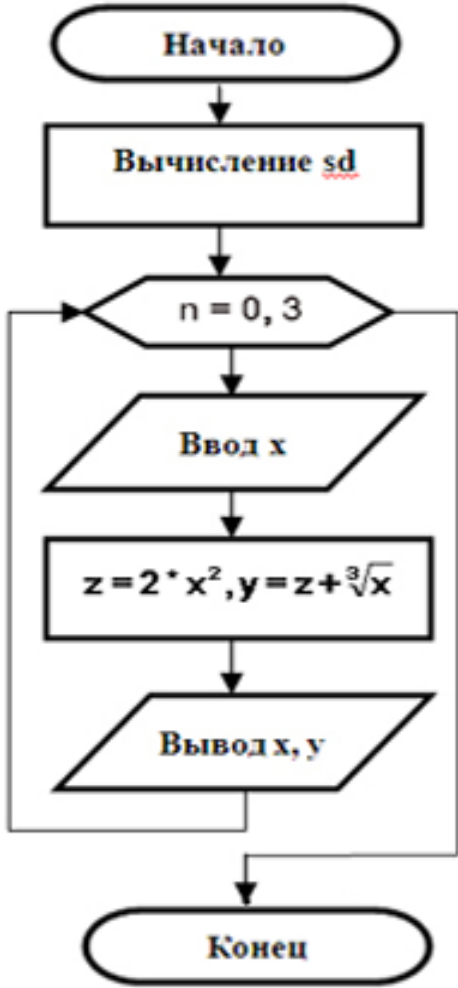
[В начало практикума](#)

Лабораторная работа № 6. Циклические программы

Цикл – это участок программы, в котором одни и те же вычисления реализуются неоднократно при различных значениях некоторых переменных, являющихся параметрами цикла.

Для организации циклов на языке C++ используются операторы: **for**, **while**, **do while**.

Задание	Краткие теоретические сведения
<p>1. Выполнить программу с использованием оператора for, записанную в правой части.</p> <p>Изучить блок-схему алгоритма.</p>	<p><u>Пример.</u> Вычислить значение y по формулам:</p> $z = 2x^2, y = z + \sqrt[3]{x},$ <p>где x = {4; 3,5; 7; 1}.</p> <p>В программе, представленной ниже, тело цикла for заключено в фигурные скобки. В переменной n находится номер значения x из массива. Вычисляется выражение n = 0 и проверяется условие n < 4. Если оно истинно, то выполняются операторы цикла.</p> <p>Затем вычисляется n = n + 1, вновь проверяется условие n < 4 и, если оно истинно, то вновь выполняются операторы цикла.</p> <p>Так продолжается до тех пор, пока n меньше 4. Если это условие не выполнится, то управление передается оператору, следующему за телом цикла.</p> <p>В переменной sd вычисляется выражение $(1 / 3)$, которое используется при вычислении корня третьей степени из x.</p> <p>Результаты выводятся в виде действительных чисел, занимают 5 позиций, из которых 2 отводятся для записи дробной части.</p>

	Блок-схема алгоритма	Программа
	 <pre> graph TD Start([Начало]) --> Calc[Вычисление sd] Calc --> Cond{n = 0, 3} Cond --> Input[/Ввод x/] Input --> Process["z = 2 * x², y = z + √³x"] Process --> Output[/Вывод x, y/] Output --> Cond Cond --> End([Конец]) </pre>	<pre> #include <iostream> void main() { setlocale(LC_CTYPE, "Russian"); float z, y, x, sd; sd = 1.0 / 3.0; for (int n = 0; n < 4; n++) { printf("Введите x "); scanf_s("%f", &x); z = 2 * pow(x, 2); y = z + pow(x, sd); printf("x = %5.2f\t", x); printf("y = %5.2f\n", y); } } </pre>

2. Выполнить программу с использованием оператора **while**, записанную в правой части.

Опробовать способ прерывания работы программы с помощью клавиш **Ctrl – Break**.

```
#include <iostream>
using namespace std;
void main()
{
    float z, y, x = 3;
    while (x < 4.1)
    {
        z = 2 * pow(x, 2);
        y = z + pow(x, (float)1 / 3);
        cout << "x=" << x << "\t";
        cout << " y=" << y << endl;
        x = x + 0.1;
    }
}
```

править ошибку.

Запись **(float)1 / 3** – это приведение типов. Тип данных в скобках указывает, в какой тип необходимо преобразовать операнд справа.

Пример. Вычислить значение **y** по формулам: $z = 2x^2$, $y = z + \sqrt[3]{x}$, где $x = 3(0,1)4$ – меняется от 3 до 4 с шагом 0,1.

В программе тело цикла **while** заключено в фигурные скобки.

Если условие в скобках (**x < 4.1**) истинно, то выполняются операторы цикла до тех пор, пока оно не станет ложным. Тогда управление передается оператору, следующему за телом цикла.

Если в программе будет отсутствовать оператор **x = x + 0.1**, то программа заиклится. Надо **прервать** выполнение программы, нажав **Ctrl – Break**, и ис-

3. Выполнить программу с использованием оператора **do while**, записанную в правой части.

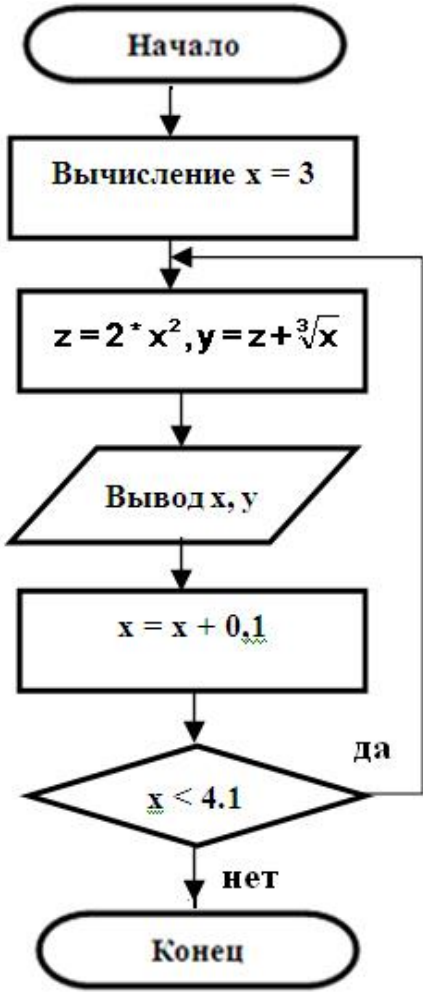
Изучить блок-схему алгоритма.

Пример. Вычислить значение **y** по формулам: $z = 2x^2$, $y = z + \sqrt[3]{x}$,

где $x = 3(0,1)4$ – меняется от 3 до 4 с шагом 0,1.

Оператор цикла **do while** называется оператором цикла с постусловием.

Сначала выполняется тело цикла, затем проверяется условие, записанное в скобках (**x < 4.1**). Если оно истинно, то выполняются операторы цикла до тех пор, пока оно не станет ложным. Тогда управление передается оператору, следующему за телом цикла.

	Блок-схема алгоритма	Программа
	 <pre> graph TD Start([Начало]) --> CalcX[Вычисление x = 3] CalcX --> CalcZY["z = 2 * x², y = z + √³x"] CalcZY --> Print["Вывод x, y"] Print --> IncX["x = x + 0.1"] IncX --> Cond{"x < 4.1"} Cond -- да --> CalcZY Cond -- нет --> End([Конец]) </pre>	<pre> #include <stdio.h> #include <cmath> void main() { float z, y, x = 3; do { z = 2 * pow(x, 2); y = z + pow(x, (float)1/3); printf("x=%5.2f\t", x); printf("y=%5.2f\n", y); x = x + 0.1; } while (x < 4.1); } </pre>

4. Выполнить программу, содержащую вложенный цикл. Записать условие задачи.

Оформить вывод результатов, используя различные возможности операторов вывода.

```
#include <stdio.h>
void main()
{
    int n, i, j;
    printf("Enter n: ");
    scanf_s("%d", &n);
    for (i = 1; i <= n; i++)
    {
        for (j = 1; j <= n; j++)
            printf("%5d", i * j);
        printf("\n");
    }
}
```

Пояснить результат.

Вместо `#include <stdio.h>` в заголовочной части записать:

```
#include <iostream>
#include <iomanip>
using namespace std;
```

Перед `printf("\n");` разместить операторы:

```
if (i == 1)
{
    cout << endl;
    cout << setw(5*n) << setfill('-') <<
        '- ' << endl;
}
```

5. В таблице приведены формулы и три варианта исходных данных, по которым надо разработать три *блок-схемы* и три *циклические* программы с одними и теми же расчетными формулами.

При наличии ошибок из-за некорректных исходных данных выполнить вычисления с другими числами.

№	Формулы для вычислений	Исходные данные (for)	Исходные данные (while)	Исходные данные (двойной цикл)
1	$h = (10r - j) / (c^2 + e^{-m})$ $y = (h \cdot m - j^2) + (0,1c)^2$	$c = 2,1; r = 4 \cdot 10^{-4}$ $m = 7;$ $j = \{4,2; 0,3; 1,7\}$	$j = 0(0,1)1,7$	$j = \{9; 1,8; 15; -3\},$ $m = 1(0,5)2$

№	Формулы для вычислений	Исходные данные (for)	Исходные данные (while)	Исходные данные (двойной цикл)
2	$y = \sqrt{i \cdot b - b^2 \cdot a}$ $z = y \cdot \operatorname{tg}(n/4) - e^{1+b}$	$a = 2 \cdot 10^{-3};$ $b = 8,5; n = 2;$ $i = \{2; 1; 8,3\}$	$i = 1(0,5)3$	$i = 2(0,5)3,$ $n = \{3; -6; 0,2; 2,8\}$
3	$z = \sqrt{a+1} - \operatorname{tg}(j+y)$ $q = e^{-j \cdot 0,1y} + (3 \cdot z)^2$	$y = -1,55; a = 6$ $j = \{8; -0,6; 1; 6,4\}$	$j = -1(0,2)1$	$a = 1(0,2)2,$ $j = \{3,3; -4; 0,9\}$
4	$y = (m \cdot j) / \operatorname{tga} - e^{10m}$ $z = 2y \cdot b + \sqrt{a+b}$	$b = 2,5; m = 3$ $a = 1,4 \cdot 10^{-3}$ $j = \{0,7; -2; 2,9\}$	$j = b(0,1)3$	$m = \{7,3; -2; 0,8\},$ $j = \{0,3; 1; 0,7; 4\}$
5	$c = i/k - \sqrt{y}/0,4$ $d = e^{1-c} + 4,9(x^2 + 1)$	$i = \{0,9; 8,4; 2\}$ $y = 1,6 \cdot 10^{-4}$ $x = -1; k = 6$	$i = 0(0,5)3$	$x = 1(0,1)2,$ $i = \{1,3; -8; 0,2\}$
6	$v = u + b - 2\sqrt{0,7k+m}$ $w = 100 \cdot e^{-0,2u} - \ln(8,1u)$	$b = -5,4; m = 4$ $u = 0,05 \cdot 10^{-4}$ $k = \{6; 4; 0,3; -7\}$	$k = 3(0,2)4$	$k = 7(0,2)8,$ $m = \{-1,3; -2; 4,9\}$
7	$z = t \cdot y^2 - \sqrt{i+x} \cdot \operatorname{tgy}$ $q = \sqrt{z^2 + 5z} \cdot \ln(y)$	$x = 0,9; y = 2;$ $t = 6,96 \cdot 10^{-5}$ $i = \{9; -1,4; 5\}$	$i = 1(0,2)2$	$i = 2(0,2)3,$ $x = \{0,7; -1; 9\}$

№	Формулы для вычислений	Исходные данные (for)	Исходные данные (while)	Исходные данные (двойной цикл)
8	$w = 0,6j + e^{t \cdot j} \cdot (4y/n)^2$ $s = \sqrt{w - 0,1t} / (2 + n^2)$	$y = 5; t = -7,4;$ $n = 9; j = \{0,5; 8,4; 0,3\}$	$j = 0(0,1)2$	$y = \{0,1; -3; 0,5\},$ $j = 0,1(0,1)0,4$
9	$y = (s + c) / \ln(f) / e^{-s}$ $h = (y - m) / \ln(y)$	$s = 5,9; m = 6$ $f = 1,2 \cdot 10^3;$ $c = \{2; 0,5; 4\}$	$c = 0(0,1)0,9$	$s = \{-3; 0,8; 4\},$ $c = 0,2(0,1)0,5$
10	$y = s / \ln(5,2f) / (e^{-s} + 2)$ $v = (1 + m \cdot y - m \cdot k) / \ln(y)$	$s = 7,4; m = 10$ $f = 3,2 \cdot 10^4;$ $k = \{4; 0,5; 8\}$	$k = 0(0,5)4$	$k = \{0,9; -11; 0,5\},$ $m = 0,3(0,1)0,7$
11	$w = \operatorname{tg}(a/3) + e^{a/m}$ $r = 0,9\sqrt{w + j} + a^2 - 1 $	$a = -1,4; m = 16$ $j = \{0,5; 9,1; 5\}$	$j = 1,8(0,2)3$	$a = \{0,2; -4; 0,6\},$ $j = 0,1(0,1)0,4$
12	$y = (m - b) / (\sin a - e^a)$ $z = 3y + \sqrt{a - 4jb}$	$b = 0,5; m = 8$ $a = 2,4 \cdot 10^4;$ $j = \{1,7; 2,9; 8\}$	$j = 2(0,5)3$	$j = 2(0,5)3,$ $m = \{0,4; -1; 1,9\}$
13	$d = \sin(k/a) / \cos(m \cdot b)$ $c = d / (d^2 + 1) / (1 - e^k)$	$a = 8; m = 6$ $b = 5 \cdot 10^3;$ $k = \{1,6; 9,1; 8\}$	$k = a(-0,5)3$	$a = 2(0,2)2,8,$ $k = \{1,7; 5; -2\}$

№	Формулы для вычислений	Исходные данные (for)	Исходные данные (while)	Исходные данные (двойной цикл)
14	$y = a / (b + e^b) / (1 + j \cdot i)$ $t = \cos(y + 1) / \sqrt{ -2j }$	$b = 2; j = 6$ $a = 1,5 \cdot 10^8;$ $i = \{7; 4; 2; 6\}$	$i = 3(-0,1)2$	$b = 2,4(0,2)3,$ $i = \{0,1; -1; 3\}$
15	$z = \sqrt{t \cdot a + y} + 4e^{-2 \cdot j}$ $w = \ln(0,4y) / (7a - z)$	$a = 1; y = 0,4;$ $t = 5 \cdot 10^5;$ $j = \{5; 3; 1; 7; 3\}$	$j = 1(0,2)2$	$t = 3(0,2)4,$ $j = \{0,2; -1,6; 9,4\}$
16	$w = \cos(a/3) + t + e^{a/m}$ $r = 0,7\sqrt{3w + j} + a^2 - 1 $	$a = 1,4; m = 6$ $t = 15 \cdot 10^5;$ $j = \{0,5; 9,1; 5\}$	$j = 1(0,1)2$	$m = 8(0,2)9,$ $j = \{0,6; -0,1; 5\}$

6. Дополнительные задания.

1. Торговая фирма в первый день работы реализовала товаров на **P** тыс. руб., а затем ежедневно увеличивала выручку на 3%. Какой будет выручка фирмы в тот день, когда она впервые превысит заданное значение **Q**? Сколько дней придется торговать фирме для достижения этого результата?

2. Фирма ежегодно на протяжении **n** лет закупала оборудование стоимостью соответственно **s1, s2, ..., sn** рублей в год (эти числа вводятся и обрабатываются последовательно). Ежегодно в результате износа и морального старения (амортизации) все имеющееся оборудование уценивается на **p**%. Какова общая стоимость накопленного оборудования за **n** лет?

3. Дана последовательность ненулевых целых чисел, которая заканчивается числом 0. Определить, сколько раз в этой последовательности меняется знак.

4. Дано натуральное число **k**. Определить **k**-ю цифру последовательности: 1248163264 ..., в которой выписаны подряд степени 2.

5. Натуральное число, записанное в десятичной системе счисления, называется сверхпростым, если оно остается простым при любой перестановке своих цифр. Определить все сверхпростые числа от 1 до n .
6. Из целого числа, введенного с клавиатуры, удалить все цифры 3 и 6 и вывести результат на экран.

Тест "Циклические программы"

[В начало практикума](#)

Лабораторная работа № 7. Отладка программ

Процесс отладки программы состоит из многократных попыток выполнения программы на компьютере, анализа полученных результатов и исправления ошибок. Отладка позволяет обнаружить ошибки, которые появляются во время *выполнения* программы, т. е. выявить ситуации, при которых программа работает, но вычисляет значения неправильно.

Средства отладки можно разделить на две группы: средства *интерактивной* отладки (позволяют выполнять программу по шагам); средства *планируемой* отладки (можно спланировать сценарий отладки на множестве шагов).

Задание	Краткие теоретические сведения	
1. Записать условие задачи для программы, представленной в правой части. Выполнить прокрутку для одной из программ лабораторной работы № 6.	<pre>#include <stdio.h> void main() { float y, x = 0; while (x < 7) { if (x < 4) y = x; else if (x > 4) y = x * x; else y = 1; printf("x=%f\t", x); printf("y=%5.2f\n", y); x = x + 2; } }</pre>	<p>Термином <i>прокрутка</i> обозначают выполнение программы вручную так, как если бы она выполнялась на компьютере.</p> <p>Запись прокрутки для данного примера может выглядеть следующим образом:</p> <p>x = 0; 0 <= 6; 0 < 4; y = x = 0; вывод x = 0 и y = 0; x = x + 2 = 0 + 2 = 2; 2 <= 6; 2 < 4; y = x = 2; вывод x = 2 и y = 2; x = x + 2 = 2 + 2 = 4; 4 <= 6; 4 = 4; y = 1; вывод x = 4 и y = 1; x = x + 2 = 4 + 2 = 6; 6 = 6; 6 > 4; y = x · x = 6 · 6 = 36; вывод x = 6 и y = 36; x = x + 2 = 6 + 2 = 8; 8 > 7. Выполнение программы закончено.</p>

<p>2. Выполнить программу из п. 1 с использованием средства <i>интерактивной</i> отладки Шаг с обходом.</p> <p>Внести изменения в программу с тем, чтобы появились ошибки (например, заменить == на = или убрать строчку x = x + 2; и т. п.). Проверить, как будет выполняться отладка по шагам с обходом.</p>	<p>Пошаговое выполнение приложения без трассировки вызываемых программ осуществляется с помощью команды Отладка / Шаг с обходом (или клавиша F10 или иконка на панели инструментов Отладка).</p> <p>При выборе отладки Шаг с обходом производится останов перед выполнением текущей строки исходного кода.</p> <p>Выполнение программы продолжается повторным нажатием клавиши F10. При этом Отладчик не заходит в используемые программой функции, такие как библиотечные, функции пользователя и т. п.</p> <p>Если поместить курсор мыши на имя переменной и удерживать его некоторое время, то появится всплывающее окно, в котором будет указано текущее значение переменной.</p> <p>В режиме отладки появляется окно Видимые, в котором отображаются имена переменных. Здесь можно изменить значение некоторой переменной, сделав двойной щелчок на отображаемом значении и записав новое значение (это средство можно использовать для коррекции программы и для пропуска первых шагов в цикле с большим количеством итераций).</p> <p>Для контроля значений удобно использовать также окно Контрольное значение (предусмотрено до 4 таких окон). Для вызова окна надо выполнить Отладка / Окна / Контрольные значения / Контрольное значение 1 (2, 3, 4) или в окне Видимые поместить курсор в столбец Имя, в контекстном меню выполнить команду Добавить контрольное значение. После появления окна надо ввести имя переменной (или выделить нужную переменную в тексте программы и перетащить ее мышкой в поле Имя).</p> <p>После отладки необходимо вывести приложение из отладочного режима командой Отладка / Остановить отладку или Shift + F5 или одноименной кнопкой на панели инструментов Отладка.</p>
--	---

<p>3. Выполнить программу из п. 1 с использованием средства <i>интерактивной</i> отладки Шаг с заходом.</p> <p>Проанализировать изменения, происходящие в процессе отладки.</p>	<p>Пошаговое выполнение приложения с трассировкой вызываемых программ осуществляется с помощью команды Отладка / Шаг с заходом (или клавиша F11, или иконка на панели инструментов Отладка).</p> <p>При этом производится останов перед выполнением текущей строки исходного кода. Выполнение программы продолжается повторным нажатием клавиши F11. При этом Отладчик заходит в используемые программой функции (библиотечные, функции пользователя и т. п.)</p>
<p>4. Выполнить программу п. 1 с использованием средств <i>планируемой</i> отладки, установив точки останова на операторах вывода и пошагово проследив вывод значений на консоль.</p> <p>Удалить точки останова.</p>	<p>При планируемой отладке используются точки останова. Точка останова назначается щелчком левой кнопки мыши в сером поле слева от строки программы (точка останова отмечается маркером в виде красного круга). Повторный щелчок на маркере точки останова приводит к ее отмене. Точку останова можно назначить также с помощью Отладка / Точка останова или клавишей F9. Она устанавливается на той строке, где помещен курсор.</p> <p>Затем приложение запускается в режиме отладки (F5) или командой меню Отладка / Продолжить. Приложение будет выполнено до точки останова.</p> <p>Продолжение вычислений – повторное нажатие клавиши F5.</p> <p>Для контроля значений можно использовать окно Локальные и окно Контрольное значение.</p> <p>В течение одной сессии отладки могут использоваться все три вышеописанных способа: шаг с заходом, шаг обходом и точки останова.</p>
<p>5. Создать исполняемый файл без отладочной информации Release.</p>	<p>После отладки и исправления всех ошибок можно построить приложение без отладочной информации. Для переключения в окончательную конфигурацию необходимо выбрать команду Построение / Диспетчер конфигураций (в новых версиях Visual Studio надо выбрать команду Сборка / Диспетчер конфигураций).</p>

<p>Объяснить назначение папок и файлов решения проекта.</p> <p>Выполнить файл с расширением *.exe в папке Debug.</p>	<p>На экран будет выведено диалоговое окно установки активной конфигурации проекта. Надо выбрать Win32 Release и повторить построение проекта с помощью Построение (или Сборка) / Перестроить проект.</p> <p>Каждая конфигурация проекта определяет папки, куда будут помещены файлы с промежуточными и окончательными результатами компиляции и компоновки. По умолчанию это папки Debug и Release, которые располагаются в папке проекта.</p> <p>Папка решения содержит файл текущего примера решения *.sln, файл с информацией о проектах решения и опциях решения *.suo. В файле решения зафиксирован перечень проектов, входящих в решение.</p> <p>Во вложенной папке находятся файлы и папки проекта: файл *.vcproj в формате XML содержит перечень файлов, включенных в проект; файл *.cpp содержит исходный код программы на языке C++; файлы *.obj – это объектные файлы, содержащие машинный код исходных файлов проекта; *.pch – предварительно скомпилированный файл заголовков; *.pdb – файл с отладочной информацией, используемой при выполнении программы в режиме отладки; *.idb – файл с информацией, необходимой для перестройки всего решения и др.</p> <p>Папка Debug используется для хранения временных файлов. В этой папке размещаются файлы с программным кодом на промежуточном языке. В частности, файл *.exe содержит программный код приложения.</p>
---	---

6. В соответствии со своим вариантом написать программу и отладить ее для задачи, представленной в таблице ниже. Опробовать средства *интерактивной* отладки **Шаг с обходом**. Вызвать окно **Контрольное значение 1**, поместить туда имена переменных и проконтролировать их значения в процессе выполнения программы.

В отчете представить результаты в окне **Отладчика** (окно **Видимые** и окно **Контрольное значение 1**).

Для той же программы представить результаты в окнах **Отладчика** для интерактивной отладки **Шаг с заходом**.

№	Формулы для вычислений	Исходные данные	№	Формулы для вычислений	Исходные данные
1	$y = \sqrt{a + m^2 \cdot x^2} / (a + x) \cdot m$ $z = \begin{cases} y + 1, & y < 1 \\ \sin^2 y, & y \geq 1 \end{cases}$	$a = 1,774;$ $m = 5;$ $x = 3(0,2)5$	6	$s = w / (w^2 - j) \cdot \ln w $ $y = \begin{cases} s - e^{f/i}, & s \geq 3\sqrt{f} \\ s^2 / w, & s < 3\sqrt{f} \end{cases}$	$f = 1,2;$ $j = -7 \cdot 10^{-3};$ $w = 2(0,5)5$
2	$d = \begin{cases} a + b \cdot i, & i > 3b \\ \operatorname{tg} b - a \cdot i, & i \leq 3b \end{cases}$ $z = (d \cdot a / 4) / (3a \cdot b - e^{1+d} / 100)$	$b = -0,05;$ $a = 1,72;$ $i = -5(2)5$	7	$w = \begin{cases} \sqrt{0,2x} \cdot k, & x < m/2 \\ e^{2 \cdot x \cdot k}, & x \geq m/2 \end{cases}$ $v = \sqrt{w^3 + x - a } / \ln(1 + a)$	$a = 1,2; k = 3$ $x = 2,5;$ $m = 4(0,2)6$
3	$s = e^{-a \cdot x} - \ln(i / x \cdot m) / \ln^2 m$ $p = \begin{cases} (-i \cdot s)^2, & s > 2x \\ \sin(-6s), & s \leq 2x \end{cases}$	$x = \{8; 1,9; -4; 1\};$ $a = 105 \cdot 10^{-4};$ $m = 4; i = 7$	8	$d = e^{-x} + (x - a) / \ln x$ $b = \begin{cases} 6d \cdot x, & d \geq k \cdot \sqrt{a} \\ \sin(a / x), & d < k \cdot \sqrt{a} \end{cases}$	$a = 1,33 \cdot 10^{-3};$ $k = 6;$ $x = \{0,2; 1; -3; -6,1\}$
4	$x = \cos^2(y) / (j + 2a \cdot y)$ $z = \begin{cases} e^{-j}, & x \geq \sqrt{y} \\ (0,5y / j)^2, & x < \sqrt{y} \end{cases}$	$y = 2,7; a = -5,5 \cdot 10^{-4};$ $j = \{6; -8,2; 15,4; 2\}$	9	$c = 2 \sin(f / 2) + \ln t$ $d = \begin{cases} y \cdot e^{-2t} + f, & c \geq 3 \\ y - y^3, & c < 3 \end{cases}$	$f = -125 \cdot 10^{-6};$ $y = 1,7; t = 1(0,1)2;$
5	$p = e^{\sin(j/x)} \cdot \ln(x/y) \cdot x$ $q = \begin{cases} \sqrt{p/m}, & p \leq y^2 \\ \sqrt{2x} / (j + p), & p > y^2 \end{cases}$	$y = 1,4 \cdot 10^{-3};$ $x = 0,2; m = 4;$ $j = 5(2)11$	10	$y = \cos^2 x^2 / x $ $z = \begin{cases} y , & y < a \cdot x \cdot n \\ \sqrt{1 + e^{-y}}, & y \geq a \cdot x \cdot n \end{cases}$	$a = 1,055; x = 0,6;$ $n = 6(2)14$

№	Формулы для вычислений	Исходные данные	№	Формулы для вычислений	Исходные данные
11	$d = i + 2t \cdot (1 + \sqrt{3a^2})$ $f = \begin{cases} t \cdot i, & d \geq t(t+i) \\ e^{t-d} + 9, & d < t(t+i) \end{cases}$	$a = -4,2;$ $t = \{5; 1; -3; 9; -1\}$ $i = 4$	14	$v = \begin{cases} \cos^2(j \cdot x), & 2s > x \cdot j \\ 2 \cdot \operatorname{tg}(j \cdot x), & 2s < x \cdot j \\ 5 - e^{x/2}, & 2s = x \cdot j \end{cases}$	$s = 5,2;$ $x = \{5; 0,4; -2,1; 6\}$ $j = 10,4 \cdot 10^4$
12	$x = \operatorname{tg}(a^2 - 1)/(d + 1)$ $y = \begin{cases} a \cdot k + d, & 3x < a \cdot c \\ \cos(a \cdot k) \cdot e^{a+1}, & 3x \geq a \cdot c \end{cases}$	$c = 3,7(0,1)5;$ $d = 51,9 \cdot 10^{-2};$ $a = 4; k = 1$	15	$b = 12s - e^{-s/2} \cdot (x - j)$ $z = \begin{cases} \sqrt{-2x \cdot j} + b, & b < 1,5 \\ 13x \cdot j + b, & b \geq 1,5 \end{cases}$	$x = -4 \cdot 10^{-3}; s = 1,1;$ $j = 4(0,5)7$
13	$t = a \cdot i / (a^2 - b) \cdot e^{-a}$ $x = \begin{cases} 4,8 \cdot 10^{-3} + i \cdot a, & t > 5c \\ a + i^2 \cdot t, & t \leq 5c \end{cases}$	$b = 3; i = 8(4)24$ $c = 6 \cdot 10^{-4}; a = 6$	16	$z = \begin{cases} \sin^2(t), & t > x \\ 4 \cdot (t + x), & t < x \\ 1 - e^{x-2}, & t = x \end{cases}$	$t = 1,5(0,1)2;$ $x = 1,9$

7. В соответствии со своим вариантом написать программу и отладить ее для задачи, представленной в таблице ниже. Опробовать средства *планируемой* отладки.

В отчете представить результаты в окне **Отладчика** (окно **Локальные** и окно **Контрольное значение 1**).

Создать *исполняемый файл* и ознакомиться с файлами в папках проекта.

№ варианта	Условие задачи
1	Используя перебор значений, вывести на экран в возрастающем порядке все трехзначные числа, в десятичной записи которых нет одинаковых цифр.
2	Найти сумму целых положительных чисел, кратных 3 и меньших 200.

№ ва- рианта	Условие задачи
3	Определить количество трехзначных натуральных чисел, сумма цифр которых равна n .
4	Вычислить наибольший общий делитель натуральных чисел a и b .
5	Построить первые N натуральных чисел, делителями которых являются только числа 2, 3 и 5.
6	Используя перебор значений вывести на экран в убывающем порядке все двузначные числа, в десятичной записи которых нет одинаковых цифр.
7	Даны натуральные числа m , n . Используя перебор значений, получить все меньшие n натуральные числа, квадрат суммы цифр которых равен m .
8	Дано натуральное число n . Получить все его натуральные делители.
9	Вывести на экран в возрастающем порядке все четырехзначные числа, в десятичной записи которых нет одинаковых цифр.
10	Используя перебор значений, найти сумму целых положительных чисел, кратных 5 и меньших 150.
11	Определить количество двузначных натуральных чисел, сумма цифр которых равна числу f .
12	Вычислить наибольший общий делитель натуральных чисел x , y и z .
13	Вывести первые 6 натуральных чисел, делителями которых являются числа 3 и 5.
14	Используя перебор значений, вывести на экран в возрастающем порядке все трехзначные числа, в десятичной записи которых нет одинаковых цифр.
15	Даны натуральные числа q и b . Используя перебор значений, получить все меньшие q натуральные числа, квадрат суммы цифр которых равен b .

№ варианта	Условие задачи
16	В заданной последовательности натуральных чисел (размер и значения вводятся пользователем) найти наибольший элемент из отрицательных.

8. Дополнительные задания.

1. Найти натуральное число, состоящее из трёх цифр, с возрастающими слева направо цифрами, являющееся полным квадратом. Число является полным квадратом, если квадратный корень из него – простое число (число 121 – полный квадрат, т. к. $121=11 \cdot 11$, а 11 – простое число).
2. Составить алгоритм, определяющий, сколько существует способов набора одного рубля при помощи монет достоинством 50 коп., 20 коп., 5 коп. и 2 коп.
3. Имеются два сосуда. В первом сосуде находится C_1 литров воды, во втором – C_2 литров воды. Из первого сосуда переливают половину воды во второй сосуд, затем из второго переливают половину в первый сосуд, и т. д. Сколько воды окажется в обоих сосудах после 12 переливаний?
4. Три приятеля были свидетелями нарушения правил дорожного движения. Номер автомобиля – четырехзначное число – никто полностью не запомнил. Из показаний следует, что номер делится на 2, на 7 и на 11, в записи номера участвуют только две цифры, сумма цифр номера равна 30. Составить алгоритм и программу для определения номера автомашины.

Тест "Отладка программ"

[В начало практикума](#)

Лабораторная работа № 8. Вычисление сумм, произведений, экстремумов

Задание	Краткие теоретические сведения
<p>1. Выполнить прокрутку и записать условие задачи для программы, представленной в правой части.</p> <p>Изменить программу так, чтобы вычислялась сумма квадратов четырех значений переменной a, а также произведение значений a.</p> <p>Произвести отладку всех вариантов.</p> <p>Написать программу вычисления факториала:</p> $5! = 1 \cdot 2 \cdot 3 \cdot 4 \cdot 5.$	<div style="border: 1px solid black; padding: 10px; margin: 10px 0;"> <pre> #include <iostream> using namespace std; void main() { setlocale(LC_CTYPE, "Russian"); float a, sum = 0; int i; const int size = 4; for (i = 0; i < size; i++) { cout << "Введите a" << i << endl; cin >> a; sum = sum + a; } cout << "Ответ " << sum << endl; } </pre> </div> <p><u>Пример.</u> Даны четыре значения переменной $a = \{0,5; 44; -8; 11\}.$</p> <p>В процессе выполнения программы они вводятся с клавиатуры по одному.</p>

2. Выполнить прокрутку и записать условия задач для программ, представленных в правой части.

```
#include <stdio.h>
#include <conio.h>

void main()
{
    char ch;
    float sv, x, sum = 0;
    float count = 0;
    do
    {
        printf("Enter x:");
        scanf_s("%f", &x);
        sum += x;
        count++;
        sv = sum / count;
        printf("sv=%1.3f\n", sv);
        printf("if continue input 'y'
else 'n' ");
        ch = _getch();
    }
    while (ch != 'n');
}
```

```
#include <stdio.h>
#include <cmath>

void main()
{
    float sum = 0, a, t, p;
    for (int n = 2; n < 10; n++)
    {
        t = pow(n, log((float)n));
        p = pow(log((float)n), n);
        a = t / p;
        sum += a;
    }
    printf("S=%f\n", sum);
}
```

3. Выполнить прокрутку и записать условие задачи для программы, представленной в правой части.

Внести изменения в программу так, чтобы выводилось не только само значение переменной из массива **b**, но и номер этого значения.

```
#include <iostream>
using namespace std;
void main()
{
    setlocale(LC_CTYPE, "Russian");
    int count = 5, n = 1;
    float b, m = 9999;
    for (int i = 0; i < count; i++)
    {
        cout << "Введите b" << i << endl;
        cin >> b;
        if (b < m)
        {
            m = b;
            n = i;
        }
    }
    cout << "m= " << m << endl;
    cout << "n= " << n << endl;
}
```

Даны пять значений переменной **b** = {9; -3,5; 54; -2,8; 40}.

4. В соответствии со своим вариантом разработать программы по условиям, приведенным в таблице ниже. Исходные данные ввести с клавиатуры. Произвести *отладку* программы.

Написать программы еще для нескольких условий из этой же таблицы по своему выбору.

№	Формулы для вычислений	Исходные данные	№	Формулы для вычислений	Исходные данные
1	$h = d + \sum_{i=1}^n a_i^2$	$n = 5; d = 12,5 \cdot 10^{-4};$ $a = \{0,8; 4; -7; 2; 0,91\}$	9	$q = t + \frac{x_1 + 1}{x_1} + \dots + \frac{x_6 + 1}{x_6}$	$t = 0,45;$ $x = \{1,1; 6,2; 3; -4; 6; 1\}$
	$z = \max\{x_i\}$ $y = z \cdot \sum_{i=1}^5 x_i^2$	$x = \{1; 2,7; 4,7; 6; 10\}$		$p = \min\{y_i\}$ $q = \prod_{i=1}^n (y_i - 5) + p$	$y = \{3; -2; 0,9; 0,5; 1\}$
2	$g = c \cdot \prod_{j=1}^m (b_j + 1)^2$	$m = 4; c = -0,045;$ $b = \{0,9; 0,5; -2; -0,1\}$	10	$q = \sum_{i=1}^n (x_i \cdot y_i)$	$n = 6;$ $x = \{3; -2; 0,7; -1; -2; 7\}$ $y = \{1; 5; -1,2; 6; 9; -4\}$
	$t = \min\{y_i\}$ $q = \prod_{i=1}^n (y_i + t)$	$y = \{6; 2; 0,9; 0,1; 5\}$		$y_i = a_i + \sum_{i=1}^5 x_i$ $k = \max\{y_i\}$	$x = \{2; 1,7; 4,7; 6; 3\}$ $a = \{0,4; 8; 15\}$
3	$z = 8 \cdot x_3 + \sum_{i=1}^m (x_i - 2)^2$	$m = 5;$ $x = \{-2; 6; 1,1; 2,7; 4\}$	11	$s = 2a + q \cdot \sin a$ $q = 4 \prod_{i=1}^n y_i / (i^2 + 1)$	$a = 5,45; n = 5;$ $y = \{2,1; 7,7; -4; 5; 9\}$
	$p = y + \max\{x_i\}$ $y = \sum_{i=1}^5 (x_i^2 + 1)$	$x = \{9; 2,7; 4,1; 6; 12\}$		$k = \prod_{i=1}^5 (y_i + \min\{y_i\})$	$y = \{1,3; 1; 0,9; 0,5; 8\}$

№	Формулы для вычислений	Исходные данные	№	Формулы для вычислений	Исходные данные
4	$f = k$ $g = f / \sum_{i=1}^5 (a_i / i)$	$k = 6(0,2)7;$ $a = \{2,3; 7; -2; -4; 9\}$	12	$z = \max \{t_i\}$ $y = t \cdot \sum_{i=1}^5 x_i^3$	$x = \{7; 2; 5,7; 6; 11\}$
	$q = \prod_{i=1}^4 (y_i + \max \{2y_i\})$	$y = \{6; 8; 0,9; 0,2; 4\}$		$q = t + \frac{x_1 + 1}{x_1} + \dots + \frac{x_6 + 1}{x_6}$	$t = 0,45;$ $x = \{1,1; 6,2; 3; -4; 6; 1\}$
5	$w_i = \begin{cases} a + v_i, & v_i > 0 \\ b / v_i, & v_i \leq 0 \end{cases}$ $c = \max \{w_i\}$	$a = 0,5; b = 7;$ $v = \{5; 2,5; 3; 8,9; 3\}$	13	$d = \min \{y_i\}$ $q = \sum_{i=1}^4 (4y_i + d)$	$y = \{5; 7; 0,9; 0,5; 2\}$
	$p = \min \{y_i\}$ $q = \prod_{i=1}^n (y_i - 5) + p$	$y = \{3; -2; 0,9; 0,5; 1\}$		$y = \begin{cases} \sum_{i=1}^5 x_i^2 + t, & t > 2 \\ \cos(t^2), & t \leq 2 \end{cases}$	$x = \{8; -2,3; -8,4; 5,1; 9\}$ $t = 0,5(0,5)3$
6	$a = \sum_{i=1}^n x_i, \quad x_i > 0$ $b = \prod_{i=1}^n x_i, \quad x_i \leq 0$	$n = 5;$ $x = \{2,7; -5; 4; 3,5; -7\}$	14	$z = \begin{cases} \sin^2 c, & \sum_{i=1}^6 y_i > c \\ \cos^2 c, & \sum_{i=1}^6 y_i \leq c \end{cases}$	$y = \{4; -6; 3; -3; 9; 5\}$ $c = 10,1$
	$c = a + \frac{x_1}{1 + x_1} + \dots + \frac{x_5}{1 + x_5}$ $a = \min \{x_i\}$	$x = \{5,1; 6,4; 3; 2; 6\}$		$m = 1 + \max \{x_i\}$ $y = \sum_{i=1}^5 (x_i^2 + m)$	$x = \{9; 1,7; 4; 6; 3\}$

№	Формулы для вычислений	Исходные данные	№	Формулы для вычислений	Исходные данные
7	$c_i = \sqrt{a_i + b_i}$ $d = \sum_{i=1}^n (c_i / i)$	$n = 6;$ $a = \{0,5; 2; 2,5; 1; 0;8\}$ $b = \{2,3; 4; 0,5; 2; 3; 9\}$	15	$q = t + \frac{x_1 + 1}{x_1} + \dots + \frac{x_6 + 1}{x_6}$	$t = 0,45;$ $x = \{1,1; 6,2; 3; -4; 6; 1\}$
	$y = \begin{cases} \min\{x_i\}, & t > 2 \\ \cos(t^2), & t \leq 2 \end{cases}$	$x = \{2; 1,3; 0,4;5,1;7\}$ $t = 0,5(0,5)3$		$c = \frac{a}{1+x_1} + \dots + \frac{a}{1+x_5}$ $a = \max\{x_i\}$	$x = \{5; 4; 3; 2; 6; 1\}$
8	$d = \sum_{i=1}^{n_1} a_i - c \cdot \sum_{i=1}^{n_2} (b_i - 1)^2$	$n_1 = 6; n_2 = 5; c = 0,7;$ $a = \{3; 12; -4; 6,2; 3; 0,4\}$ $b = \{19; 1; -24; 4,2; 8\}$	16	$y = \begin{cases} \sum_{i=1}^5 x_i^2 + t, & t > 2 \\ \cos(t^2), & t \leq 2 \end{cases}$	$x = \{8; -2,3; -8,4;5,1;9\}$ $t = 0,5(0,5)3$
	$z = \min\{(2 + x_i)\}$ $y = \sum_{i=1}^5 (x_i^2) + z$	$x = \{1; 6,7; 4; 6; 17\}$		$z = \begin{cases} \max\{x_i\}, & t > 1 \\ \sin(t^2), & t \leq 1 \end{cases}$	$x = \{4; 0,3;4;1;7\}$ $t = 0,6(0,2)2$

6. Дополнительные задания.

1. В последовательности из **n** целых чисел найти и вывести значение суммы четных элементов.
2. В последовательности из **n** целых чисел найти и вывести порядковый номер последнего отрицательного элемента.
3. В последовательности из **n** вещественных чисел найти количество элементов, стоящих между минимальным и максимальным значениями.

4. В последовательности найти число чередований знака, т. е. число переходов с минуса на плюс или с плюса на минус. Пример: в последовательности целых чисел 0, −2, 0, −10, 2, −1, 0, 0, 3, 2, −3 четыре чередования (как известно, нуль не имеет знака).
5. Последовательность a_1, a_2, \dots, a_k называется пилообразной, если $a_1 < a_2 > a_3 < a_4 > \dots > a_k$ либо $a_1 > a_2 < a_3 > a_4 < \dots < a_k$. Проверить является ли заданная последовательность пилообразной.

Тест "Суммы, произведения, экстремумы"

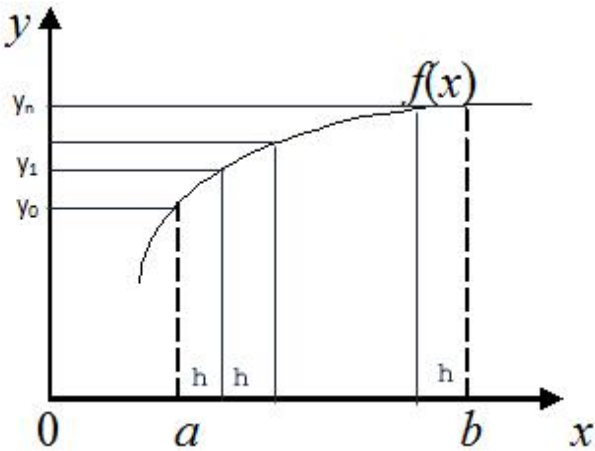
[В начало практикума](#)

Лабораторная работа № 9. Решение инженерных задач на основе циклических программ

В инженерной практике часто встречаются задачи, для решения которых требуется использовать численные методы. Численные методы или методы вычислительной математики характеризуются тем, что решение конкретной задачи сводится к выполнению арифметических действий.

Рассмотрим способы использования циклических алгоритмов для вычисления площади криволинейной трапеции и для решения уравнений.

Задание	Краткие теоретические сведения
1. Изучить алгоритм вычисления <i>площади</i> криволинейной трапеции методом <i>трапеций</i> .	<p>Пусть требуется найти площадь криволинейной трапеции, ограниченной графиком функции $f(x)$, прямыми $x = a = x_0$, $x = b = x_n$ и осью OX.</p> <p>Площадь можно приближенно вычислить, разбив ее на части и вычислив сумму площадей получившихся фигур.</p> <p>Интервал $[a, b]$ надо разделить на n равных частей длиной $h = (b - a) / n$. Тогда значениям $x_i = x_{i-1} + h, i = 1, 2, ..., n$ соответствуют значения $y_i = f(x_i)$.</p> <p>Согласно <i>методу трапеций</i> значение площади всей фигуры вычисляется как сумма площадей трапеций, высоты которых равны h, а основания соответственно y_0 и y_1 – для первой трапеции, y_1 и y_2 – для второй и т. д.</p> <div>$S \approx h \frac{y_0 + y_1}{2} + h \frac{y_1 + y_2}{2} + ... + h \frac{y_{n-1} + y_n}{2} = \frac{h}{2} (y_0 + 2(y_1 + y_2 + ... + y_{n-1}) + y_n)$</div>



	<p>Алгоритм вычисления площади по методу трапеций:</p> <ol style="list-style-type: none"> 1. Ввод a, b, n. 2. Вычисление h = (b - a) / n; x = a; s = 0. 3. Расчет s = s + h · (f(x) + f(x + h)) / 2, x = x + h. 4. Если x > (b - h), то переход к пункту 5, иначе – переход к пункту 3. 5. Вывод s.
<p>2. Изучить алгоритм вычисления площади криволинейной трапеции методом <i>парабол</i>.</p>	<p>При использовании <i>метода парабол</i> интервал [a, b] делится на четное количество частей – 2n. Тогда h = (b - a) / (2·n); x_i = x_{i-1} + h; i = 1, 2, ..., 2n.</p> $S \approx \frac{h}{3} (y_0 + 4(y_1 + y_3 + \dots + y_{2n-1}) + 2(y_2 + y_4 + \dots + y_{2n-2}) + y_{2n}).$ <p>Алгоритм метода парабол:</p> <ol style="list-style-type: none"> 1. Ввод a, b, n. 2. Вычисление h = (b - a) / (2·n); x = a + 2h; s1 = 0; s2 = 0; i = 1. 3. Расчет s2 = s2 + f(x); x = x + h; s1 = s1 + f(x); x = x + h; i = i + 1. 4. Если i < n, то переход к пункту 3, иначе – переход к следующему пункту. 5. Вычисление значения площади криволинейной трапеции: $S = \frac{h}{3} (f(a) + 4f(a + h) + 4s1 + 2s2 + f(b))$ <ol style="list-style-type: none"> 6. Вывод S. <p>Здесь s1 = y₃ + y₅ + ... + y_{2n-1}, s2 = y₂ + y₄ + ... + y_{2n-2}.</p>

<p>3. Изучить графический способ отделения корней и <i>метод дихотомии</i> для решения уравнения.</p>	<p>Решение уравнений численными методами состоит из двух этапов: <i>отделение</i> корней, т. е. нахождение таких отрезков $[a, b]$ на оси ОХ, внутри которых имеется один корень; <i>вычисление</i> корней с заданной точностью.</p> <p><i>Графический метод</i> отделения корней: в исходном уравнении надо перенести в правую часть некоторые члены так, чтобы легко можно было построить два графика функций левой и правой частей. Точки пересечений графиков определяют примерное расположение корней.</p> <p>Для вычисления корня уравнения $f(x) = 0$ на отрезке $[a, b]$ можно использовать <i>метод дихотомии</i>, согласно которому отрезок $[a, b]$ делится пополам. Из полученных двух отрезков выбирается тот, на концах которого функция $f(x)$ имеет разные знаки. Выбранный отрезок вновь делится пополам. Вычисления продолжаются до тех пор, пока величина последнего из полученных отрезков не станет меньше $2e$, где e – точность вычислений.</p> <p>Алгоритм метода дихотомии:</p> <ol style="list-style-type: none"> 1. Ввод значений a, b, e. 2. Вычисление $x = (a + b) / 2$. 3. Если $f(x)f(a) \leq 0$, то $b = x$, иначе – $a = x$. 4. Если $a - b > 2e$, то переход к пункту 2, иначе – переход к следующему пункту. 5. Вывод значения корня x.
<p>4. Изучить способ нахождения корня уравнения в приложении Excel с помощью команды Подбор параметра.</p>	<p>В приложении Excel для решения уравнения имеется команда Подбор параметра.</p> <p>Чтобы найти корень уравнения (например, $4 - x^2 + x = 0$) надо на рабочем листе, например, в ячейке A1, записать начальное приближение корня, в ячейке B1 – уравнение: $= 4 - A1^2 + A1$</p> <p>Выполнить Данные / Работа с данными / Анализ “что-если” / Подбор параметра.</p> <p>В появившемся окне задать следующие значения: в поле Установить в ячейке выбрать B1, в поле Значение ввести 0, в поле Изменяя значение ячейки – A1.</p> <p>После нажатия ОК в ячейке A1 будет корень уравнения.</p>

5. В соответствии со своим вариантом написать программы для вычисления площади криволинейной трапеции по исходным данным из таблицы, приведенной ниже, методом *трапеций* и методом *парабол*. Для всех вариантов принять $n = 200$. Сравнить результаты, которые должны отличаться на небольшую величину.

№ ва- рианта	Функция f(x)	Границы от- резков	№ ва- рианта	Функция f(x)	Границы от- резков
1	$x^3 - 3$	$a = 1, b = 3$	9	$x^3 + 3$	$a = 3, b = 6$
2	$\cos^3(x)$	$a = 4, b = 7$	10	$x^2 - 4$	$a = 4, b = 8$
3	$1 + x^3$	$a = 1, b = 6$	11	$\sin(x) + 1$	$a = 1, b = 3$
4	$e^x - 1 / x$	$a = 2, b = 3$	12	$e^x + 6$	$a = 5, b = 11$
5	$5 - x^2$	$a = 8, b = 12$	13	$2 + x^3$	$a = 8, b = 14$
6	$1 + x^3$	$a = 1, b = 5$	14	$x^4 + 4$	$a = 1, b = 4$
7	$e^x + 2$	$a = 5, b = 11$	15	$\sin^2(x) + 1$	$a = 2, b = 7$
8	$x^3 - 1$	$a = 0, b = 3$	16	$x^2 + 1 / x$	$a = 0, b = 4$

Площадь криволинейной трапеции – это значение определенного интеграла $\int_a^b f(x)dx$. Для конкретной задачи его значение можно проверить в приложении Mathcad.

6. В соответствии со своим вариантом найти отрезок (значения **a** и **b**), который содержит один корень, *отделив корни* уравнения *графическим* методом для исходных данных из таблицы, приведенной ниже. Если корней несколько, то выбрать один из отрезков.

Написать программу вычисления корня уравнения методом *дихотомии*. Точность вычислений принять равной $\epsilon = 0,0001$ для всех вариантов.

Найти корень уравнения с помощью приложения Excel.

Сравнить результаты.

№ варианта	Функция $f(x)$	№ варианта	Функция $f(x)$	№ варианта	Функция $f(x)$
1	$x^3 + x - 3$	7	$x^3 + 3x - 1$	13	$e^x + x - 4$
2	$\cos(x) + x - 7$	8	$x^3 + x - 4$	14	$x^3 + x - 2$
3	$x^3 + 2x - 1$	9	$\sin(x) + x^3$	15	$\sin(x) + 2 + x$
4	$e^x - 3 - 1/x$	10	$e^x + 2x^2 - 3$	16	$x^2 + 4x - 2$
5	$2 - x^2 + x$	11	$2x + x^3 - 7$		
6	$5x - 1 + x^3$	12	$x^3 + 2x - 4$		

Контрольная работа¹ 1

[В начало практикума](#)

Лабораторная работа № 10. Одномерные массивы

Массив – это группа элементов одного типа (**double**, **float**, **int** и т. п.) последовательно расположенных в памяти и объединенных одним именем. Пример объявления массива в программе: **int a[6];**
Нумерация элементов массива начинается с **нуля** и заканчивается **n – 1**, где **n** – число элементов массива.

Задание	Краткие теоретические сведения	
<p>1. Выполнить программу, записанную в правой части. Опробовать второй вариант генерации чисел, записанный в комментарии.</p> <p>Добавить в программу операторы вычисления суммы элементов массива A.</p> <p>Произвести отладку.</p>	<pre>#include <iostream> #include <ctime> void main() { setlocale(LC_CTYPE, "Russian"); using namespace std; const int N = 100; int i, sz, A[N], rmn = 0, rmx = 99; cout << "Введите размер массива "; cin >> sz; cout << "Массив A:" << endl; srand((unsigned)time(NULL)); for (i = 0; i < sz; i++) { A[i] = (int)(((double)rand() / (double)RAND_MAX) * (rmx - rmn) + rmn); cout << A[i] << endl; } }</pre>	<p><u>Пример.</u> Сформировать одномерный статический массив целых чисел A, используя датчик случайных чисел (в диапазоне от 0 до 99).</p> <p>В программе массив заполняется случайными числами с помощью функции rand, которая генерирует целое число в диапазоне от 0 до RAND_MAX (константа, равная 32767).</p> <p>При каждом запуске программы будут генерироваться одни и те же случайные числа.</p> <p>Чтобы числа были разными нужно использовать функцию srand, которая задает начальное число с помощью функции time,</p>

возвращающей текущее время в секундах.

	<p>При использовании функции time нужно включить в заголовок директиву:</p> <p>#include <ctime> или #include <locale></p>
<p>2. Выполнить программу, записанную в правой части. Записать ее условие.</p> <p>Добавить в программу операторы вычисления среднего значения элементов исходного массива.</p> <p>Произвести отладку.</p>	<div><pre>#include <iostream> using namespace std; void main() { setlocale(LC_CTYPE, "Russian"); int i, k, size = 4; float massivA[] = {5, -4, 17.1, 9, 1}; cout << "Введите номер элемента (от 0 до 4)"; cin >> k; for (i = k; i <= size; i++) massivA[i] = massivA[i + 1]; size--; for (i = 0; i <= size; i++) cout << massivA[i] << endl; }</pre></div> <p>Инициализация массива означает присвоение начальных значений его элементам при объявлении.</p> <p>Массивы можно инициализировать списком значений, заключенных в фигурные скобки, например,</p> <p>float massivA[] = {5, -4, 17.1, 9, 1};</p> <p>Тогда длина массива вычисляется компилятором по количеству значений, перечисленных в фигурных скобках.</p>
<p>3. Выполнить программу, приведенную в правой части.</p>	<p><u>Пример.</u> Сформировать массив целых чисел в количестве не более 30. Размерность массива ввести с клавиатуры.</p> <p>Найти в массиве наибольший элемент.</p>

Внести изменения с тем, чтобы вычислялся минимальный элемент массива.

Произвести отладку.

```
#include <locale>
#include <iostream>

void main()
{
    setlocale(LC_ALL, "rus");
    using namespace std;
    const int maxSize = 30;
    int n, i, kmax = 0;
    int a[maxSize];
    cout << "Введите размер массива (<30)" << endl;
    cin >> n;
    if (n > 30)
        return;
    srand((unsigned)time(NULL));
    for (i = 0; i < n; i++)
    {
        a[i] = rand() % 30;
        cout << a[i] << " ";
    }
    cout << endl;
    for (i = 1; i < n; i++)
        if (a[i] > a[kmax])
            kmax = i;
    cout << "Максимальный элемент " << a[kmax] << endl;
}
```

4. Выполнить прокрутку программы, приведенной в правой части. Записать условие.

Опробовать программу для массивов **A** и **B** разного размера.

```
#include <stdio.h>
void main()
{
    const int size = 5;
    int masA[size], masB[size], masC[size*2];
    int k = 0, j = 0, i = 0;
    printf("A:\n");
    for (int n = 0; n < size; n++)
        scanf_s("%d", &masA[n]);
    printf("B:\n");
    for (int n = 0; n < size; n++)
        scanf_s("%d", &masB[n]);
    do
    {
        if (masA[k] <= masB[j])
            masC[i++] = masA[k++];
        else
            masC[i++] = masB[j++];
        if (k == size)
            for (; j < size; j++)
                masC[i++] = masB[j];
        if (j == size)
            for (; k < size; k++)
                masC[i++] = masA[k];
    }
    while (i < size * 2);
    printf("\n");
    for (i = 0; i < size * 2; i++)
        printf("%d ", masC[i]);
    printf("\n");
}
```

Пример. Даны два массива целых чисел **A** и **B** размера 5, элементы которых предварительно упорядочены по возрастанию. В программе сформирован массив **C**.

Определить, упорядочен ли этот массив.

5. В соответствии со своим вариантом написать программы по условиям, представленным в таблице ниже. Использовать одномерный массив целых случайных чисел (диапазон от 0 до 99). Размер массива ввести с клавиатуры.

Представить результаты в окне **Отладчика**.

Для одной из программ составить *блок-схему* алгоритма.

№ варианта	Условие задачи
1	<p>1. Удалить все элементы с заданным значением, если они имеются в массиве. Добавить перед каждым четным элементом массива элемент со значением 1.</p> <p>2. Задан массив из k символов. Создать два новых массива: в первый перенести все цифры из исходного массива, во второй – все остальные символы.</p>
2	<p>1. Все четные элементы целочисленного массива K(n) поместить в массив L(n), а нечетные – в массив M(n). Подсчитать количество тех и других.</p> <p>2. Задан массив из k чисел. Сдвинуть элементы массива циклически на n позиций вправо.</p>
3	<p>1. Удалить элементы, индексы которых кратны 7. Добавить после каждого нечетного элемента массива элемент со значением 4.</p> <p>2. Задан массив из k чисел. Преобразовать массив следующим образом: все отрицательные элементы массива перенести в начало, а все остальные – в конец.</p>
4	<p>1. Удалить элемент с номером k. Добавить после каждого четного элемента массива элемент со значением 0.</p> <p>2. Задан массив из n символов. Создать два новых массива: в один массив перенести все цифры из исходного массива, в другой – все остальные символы.</p>

№ ва- рианта	Условие задачи
5	<p>1. В массиве C каждый третий элемент заменить полусуммой двух предыдущих.</p> <p>2. Задан массив из n символов. Определить, симметричен ли он, т. е. читается ли он одинаково слева направо и справа налево.</p>
6	<p>1. Поменять местами минимальный и максимальный элементы массива.</p> <p>2. Задано два массива. Найти наименьшие среди элементов первого массива, которые не входят во второй массив.</p>
7	<p>1. Удалить из массива все элементы, совпадающие с его минимальным значением. Добавить в начало массива три элемента со значением, равным среднему арифметическому массива.</p> <p>2. Задан массив из k символов. Расположить в массиве символы в обратном порядке.</p>
8	<p>1. Найти максимальный элемент массива и заменить им нечетные по номеру элементы.</p> <p>2. Задан массив из k чисел. Определить количество инверсий в массиве (т. е. таких пар элементов, в которых большее число находится слева от меньшего).</p>
9	<p>1. Найти в массиве элемент, наиболее близкий к среднему арифметическому суммы его элементов.</p> <p>2. Задано два массива. Найти наибольшие среди элементов первого массива, которые не входят во второй массив.</p>
10	<p>1. Найти в массиве элемент, если он существует, равный среднему арифметическому суммы трех его последних элементов.</p> <p>2. Задан массив из k чисел. Найти числа, входящие в массив только один раз.</p>

№ ва- рианта	Условие задачи
11	<p>1. Удалить пять первых нечетных элементов массива. Добавить в конец массива три новых нулевых элемента.</p> <p>2. Задан массив из k символов. Удалить из него повторные вхождения каждого символа.</p>
12	<p>1. Найти минимальный элемент массива T и заменить им четные по номеру элементы.</p> <p>2. Задан массив из k символов. Определить количество различных элементов в массиве.</p>
13	<p>1. В массиве A каждый элемент, кроме первого, заменить суммой всех предыдущих.</p> <p>2. Задан массив из k символов латинского алфавита. Вывести на экран все символы, которые входят в этот массив по одному разу.</p>
14	<p>1. В массиве несколько нулевых элементов. Найти первый и последний нулевые элементы. Вывести их индексы.</p> <p>2. Задан массив из k чисел. Найти число, наиболее часто встречающееся в этом массиве.</p>
15	<p>1. Удалить элементы, индексы которых кратны 3. Добавить после каждого отрицательного элемента массива элемент со значением 10.</p> <p>2. Задан массив из k чисел. Сдвинуть элементы массива циклически на n позиций влево.</p>
16	<p>1. В массиве найти первый минимальный и первый максимальный элементы. Вывести их индексы.</p> <p>2. Задан массив из k чисел. Найти числа, входящие в массив только два раза.</p>

7 . Дополнительные задания.

1. Имеются результаты n ежедневных измерений количества выпавших осадков. За какую из недель (отрезок времени длиной 7 дней), считая с начала периода измерений, выпало наибольшее количество осадков?
2. Подсчитать количество пар соседних элементов массива с одинаковыми значениями.
3. Найти в массиве наибольшее число подряд идущих одинаковых элементов.
4. В массиве M , размером k , много совпадающих элементов. Найти количество различных элементов в нем (не упорядочивая массив).
5. В массиве A , размером $2n + 1$, не содержащем одинаковых элементов, найти средний по величине элемент, т. е. такой, при котором в массиве A ровно n элементов меньше его и столько же элементов больше его. Массив A сохранить (не сортировать), дополнительных массивов не использовать.
6. Найти непрерывную последовательность положительных чисел, сумма элементов в которой максимальная. Максимальный размер массива 100 элементов. Диапазон значений от -100 до 100.

Тест "Одномерные массивы"

[В начало практикума](#)

Лабораторная работа № 11. Битовые операции

На языке C/C++ предусмотрены битовые операции для работы с отдельными битами. Их **нельзя** применять к переменным **вещественного** типа. Основные битовые операции:

- **AND (и)** – **&** – если какой-то бит в одном из операндов равен 0, то результирующий бит тоже будет равен 0;
- **OR (или)** – **|** – если какой-то бит в одном из операндов равен 1, то результирующий бит тоже будет 1;
- **XOR (исключающее или)** – **^** – результирующий бит равен 1, если сравниваемые биты различны;
- **NOT (не)** – **~** – меняются все биты на противоположные;
- **сдвиг влево** – **<<** – удваивается значение; – **сдвиг вправо** – **>>** – значение уменьшается в два раза.

Задание	Краткие теоретические сведения	
1. Изучить использование битовых операций и маскирования числа, опробовав программу, записанную в правой части, с различными исходными числами.	<pre>#include <iostream> using namespace std; void main() { setlocale(LC_CTYPE, "Russian"); unsigned int value; int i; const unsigned int mask = 1 << 31; cout << "Введите целое число "; cin >> value; cout << "Двоичный вид: "; for (i = 1; i <= 32; i++) { putchar(mask & value ? '1' : '0'); value <<= 1; if (i % 8 == 0) putchar(' '); } }</pre>	<p><u>Пример</u> программы, печатающей тридцатидвухразрядное двоичное представление целого числа, введенного с клавиатуры.</p> <p>Здесь используется маскирование всех битов числа, за исключением текущего, выводимого на печать.</p> <p>Маска имеет вид:</p> <div>10000000 00000000 00000000</div> <div>00000000, т.е. 1 << 31.</div> <p>Последовательно применяется маска и сдвигается число на разряд влево.</p>

<p>2. Выполнить программу, записанную в правой части. Ознакомиться с результатом.</p> <p>Опробовать программу, изменяя различные биты различных чисел.</p>	<pre>#include <iostream> using namespace std; void main() { setlocale(LC_CTYPE, "Russian"); int A = 150; char tmp[33]; _itoa_s(A, tmp, 2); cout << " Число A: " << tmp << endl; _itoa_s(0x24, tmp, 2); cout << " Маска для A: " << tmp << endl; _itoa_s(A 0x24, tmp, 2); cout << " Результат: " << tmp << endl<<endl; }</pre>	<p><u>Пример.</u> Установить в единицу каждый третий по счету бит числа A считая справа.</p> <p>Здесь для вывода двоичного представления числа используется стандартная функция:</p> <p><u>_itoa_s</u> (число ввода, строка вывода, основание системы счисления).</p>
<p>3. В программе, записанной в правой части, используются различные битовые операции.</p> <p>Внести изменения в программу с тем, чтобы проверялось число на кратность четырем.</p>	<pre>#include <iostream> using namespace std; void main() { setlocale(LC_CTYPE, "Russian"); int A; char tmp[33]; cout << "Введите число "; cin >> A; _itoa_s(A, tmp, 2); cout << "Число в двоичном виде = "<<tmp<<endl; if ((A & 7) == 0) cout << "Число кратно 8" << endl; else cout << "Число не кратно 8" << endl; }</pre>	<p><u>Пример.</u> Пусть имеется некоторое целое число. Вывести его двоичное представление и проверить, кратно ли оно восьми.</p>

4. В правой части приведен пример программы, демонстрирующей использование битовых операций.

Проанализировать текст программы и написать пояснения.

```
#include <iostream>
using namespace std;

void main()
{
    setlocale(LC_CTYPE, "Russian");
    char tmp[33];
    int A, B, maskA = 14;
    int maskB = ~maskA >> 1;
    cout << "Первое число A="; cin >> A;
    cout << "Второе число B="; cin >> B;
    _itoa_s(A, tmp, 2);
    cout << "A=" << tmp << endl;
    _itoa_s(B, tmp, 2);
    cout << "B=" << tmp << endl;
    _itoa_s(maskA, tmp, 2);
    cout << "Маска для A: " << tmp << endl;
    _itoa_s((A & maskA) >> 1, tmp, 2);
    cout << "Выделенные биты A: " << tmp << endl;
    _itoa_s(maskB, tmp, 2);
    cout << "Маска для B: " << tmp << endl;
    _itoa_s(B & maskB, tmp, 2);
    cout << "Очищены биты в B: " << tmp << endl;
    _itoa_s(((B & maskB) | ((A & maskA) >> 1)), tmp, 2);
    cout << "Результат B=" << tmp << endl;
}
```

Пример. Извлечь 3 бита числа А, начиная со второго по счету справа и вставить их в число В, начиная с первого по счету справа.

5. В соответствии со своим вариантом разработать программы, использующие *битовые* операции для решения задач, представленных в таблице.

Результаты одной из программ представить в **Отладчике**.

№ варианта	Условия задач
1	<ol style="list-style-type: none"> 1. Ввести целое A и посчитать, сколько нулей в числе начиная с третьего бита по 13, включая эти биты. 2. Инвертировать в числе A n битов вправо от позиции p, заменить ими m битов числа B, начиная с позиции q.
2	<ol style="list-style-type: none"> 1. Извлечь 5 битов числа A, начиная со второго и вставить их в число B, начиная с третьего бита. 2. Установить в 1 в числе A n битов вправо от позиции p.
3	<ol style="list-style-type: none"> 1. Ввести целое число A. Инвертировать все биты с 2 по 14, включая эти биты. Вывести результат. 2. Инвертировать n битов в числе A влево от позиции p, заменить ими m битов числа B, начиная с позиции q.
4	<ol style="list-style-type: none"> 1. Используя битовые операции проверить, кратно ли четырем число A. 2. Установить в 1 в числе A n битов влево от позиции p.
5	<ol style="list-style-type: none"> 1. Определить, насколько в числе A больше значащих битов, равных единице, чем битов, равных нулю. 2. Установить в 1 n битов в числе A вправо от позиции p, заменить ими m битов числа B, начиная с позиции q.
6	<ol style="list-style-type: none"> 1. Установить в единицу каждый второй значащий бит целого числа A. 2. Извлечь 3 бита числа A, начиная с позиции n, и вставить в число B, начиная с позиции m.

№ варианта	Условия задач
7	<ol style="list-style-type: none"> 1. Извлечь 4 бита числа A, начиная с пятого по счету справа, и добавить их к числу B справа. 2. Установить в 1 n битов в числе A влево от позиции p, заменить ими m битов числа B, начиная с позиции q.
8	<ol style="list-style-type: none"> 1. Установить в ноль каждый третий значащий бит целого числа A. 2. Извлечь 3 бита числа A, начиная с позиции n, и вставить в число B, начиная с позиции m.
9	<ol style="list-style-type: none"> 1. Извлечь 5 битов числа A, начиная с третьего по счету справа, и вставить их в число B, начиная со 2 по счету справа. 2. Установить в 0 n битов в числе A вправо от позиции p.
10	<ol style="list-style-type: none"> 1. Вывести 6 бит целого числа A, начиная со 2-ого n битов. 2. Инвертировать n битов в числе A влево от позиции p.
11	<ol style="list-style-type: none"> 1. Используя битовые операции проверить, кратно ли шестнадцати число A. 2. Установить в 0 n битов в числе A влево от позиции p, заменить ими m битов числа B, начиная с позиции q.
12	<ol style="list-style-type: none"> 1. Ввести целое число A. Инвертировать все биты с 4 по 8, включая эти биты. Вывести полученное число. 2. Установить в 0 n битов в числе A вправо от позиции p, заменить ими m битов числа B, начиная с позиции q.
13	<ol style="list-style-type: none"> 1. Ввести целое число A. Извлечь 2 бита числа A, начиная с пятого по счету справа, и вставить их в число B, начиная также с пятого бита по счету справа. 2. Инвертировать в 1 n битов в числе A вправо от позиции p.

№ варианта	Условия задач
14	<ol style="list-style-type: none"> 1. Ввести целое число A и посчитать, сколько единиц в числе с 5 по 10 бит, включая эти биты. 2. Извлечь 3 бита числа A, начиная с позиции n по счету справа, и вставить в число B, начиная с позиции m.
15	<ol style="list-style-type: none"> 1. Используя битовые операции проверить, кратно ли двум число A. 2. Установить в 0 n битов в числе A влево от позиции p.
16	<ol style="list-style-type: none"> 1. Ввести целое число A. Извлечь 3 бита числа A, начиная со второго по счету справа, и вставить их в число B, начиная с первого бита по счету справа. 2. Установить в 1 два бита числа A, начиная с четвертого по счету справа.

Тест "Битовые операции"

[В начало практикума](#)

Лабораторная работа № 12. Использование указателей

Указатель – это переменная, значением которой является **адрес ячейки памяти**, в которой содержится значение другой переменной. В памяти компьютера указатель обычно занимает 4 байта.

Задание	Краткие теоретические сведения	
<p>1. Изучить работу с <i>указателями</i>, выполнив программы, записанные в правой части.</p> <p>Использовать средства отладки, проанализировать память компьютера.</p> <p>Записать условия задач.</p>	<p>Пусть объявлена переменная и указатель на нее: int a = 0; int *ptr; Можно в указатель поместить адрес переменной и вывести его: ptr = &a; cout << ptr;</p> <p>Здесь & – операция получения адреса переменной, * – операция разыменования (получения значения переменной, на которую указывает указатель).</p>	
	<pre>#include <iostream> using namespace std; void main() { int a = 10, *pa, b = 20, *pb; pa = &a; cout << &a << " " << a << endl; cout << pa << " " << *pa << endl; pb = &b; cout << pb << " " << *pb << endl; *pa = *pb; cout << &a << " " << a << endl; }</pre>	<pre>#include <iostream> void main() { using namespace std; float a, *pa, s = 0; int i; pa = &a; for (i = 1; i <= 4; i++) { cout << "a="; cin >> a; s += *pa; } cout << "s=" << s << endl; }</pre>

При использовании средств отладки в окне **Видимые** или **Локальные** или **Контрольное значение** рядом с именем переменной может стоять знак плюс. После щелчка по символу «+» появляется значение, хранимое в памяти по адресу, содержащемуся в указателе.

Для изучения расположения данных в памяти при использовании указателей полезно проконтролировать содержимое памяти компьютера с помощью **Отладка / Окна / Память / Память 1** (2, 3, 4). Вывод адресов и содержимого переменных производится в *шестнадцатеричной* системе счисления.

2. В правой части записаны фрагменты программ с использованием *указателя на константу, константного указателя, константного указателя на константу.*

Убрать ошибочные операторы, дописать операторы вывода и выполнить программы на компьютере.

Написать программу генерации элементов массива *A* из случайных чисел, их вывода и определения максимального элемента массива двумя способами (с указателями и без них).

Указатель на константу

```
{ int a = 9;
  const int *pa = &a;
  *pa = 12; // ошибка
(нельзя менять значение переменной a)
  int b = 20;
  pa = &b;
}
```

Константный указатель

```
{  int a = 99;
    int* const pa = &a;
    int b = 44;
    pa = &b; // ошибка
    (константный указатель ме-
    нять нельзя)
    *pa = 12;
}
```

Константный указатель на константу

```
{   int a = 99;
    const int* const pa = &a;
    *pa = 33; // ошибка (нельзя
              // менять содержимое
              // переменной a)
    int b = 44;
    pa = &b; // ошибка (кон-
              // стантный указатель
              // на константу
              // менять нельзя)
}
```

Имя массива **A** без индекса является **указателем-константой** (не изменяется на протяжении всей работы программы), т. е. **адресом первого** элемента массива **A[0]**.

Цикл, в котором генерируется и выводится массив **A**, содержащий **size** элементов (случайные числа от 0 до **n – 1**), можно реализовать тремя способами:

[illegible]

<p>3. Выполнить программу, приведенную в правой части, которая разработана с использованием указателей.</p> <p>Внести изменения с тем, чтобы программа стала содержать ошибки. Исследовать их с помощью отладки.</p>	<pre>#include <iostream> void main() { setlocale(LC_CTYPE, "Russian"); using namespace std; int i, k, sz = 5; float A[] = { 5, -4, 17.1, 9, 1 }; cout << "Введите номер элемента (от 0 до 4) " << endl; cin >> k; cout << endl; for (i = k; i < sz - 1; i++) *(A + i) = *(A + i + 1); sz--; for (i = 0; i < sz; i++) cout << *(A + i) << endl; }</pre>	<p><u>Пример.</u> Значения элементов массива A инициализируются в программе. Удалить элемент с номером, который вводится с клавиатуры.</p>
<p>4. Изучить отличия ссылок от указателей. Выполнить программы, записанные в правой части.</p>	<p>Ссылку (ссылочный тип) можно рассматривать как <i>альтернативное</i> имя переменной или как указатель, который всегда разыменовывается. Например, int kol; int &pal = kol;</p> <p>Здесь & – оператор ссылки, означающий, что следующее за ним имя является именем переменной ссылочного типа, pal – альтернативное имя для kol.</p> <p>Между ссылкой и указателем существуют два основных отличия:</p> <ul style="list-style-type: none">– ссылка обязательно должна быть инициализирована в месте своего определения;– всякое изменение ссылки преобразует не ее, а тот объект, на который она ссылается.	

		Использование ссылки	Переход от ссылки к указателю
		<pre>#include <iostream> void main() { int val = 1; std::cout<<"value = "<<val; int &rv = val; rv = 5; std::cout<<" value = "<<val; }</pre>	<pre>#include <iostream> void main() { int val = 1; std::cout<<"value = "<<val; int &rv = val; rv = 5; int *pval = &rv; std::cout<<" value = "<< *pval; }</pre>

5. В соответствии со своим вариантом написать программы для условий задач из таблицы, представленной ниже, с использованием *указателей* для доступа к элементам массива. Проанализировать содержимое памяти компьютера.

№ ва- рианта	Условие задачи
1	1. Ввести целое число N . Выделить из этого числа цифры, кратные m , и записать их в одномерный массив. 2. Заданы два массива по 10 целых чисел в каждом. Найти наибольшее среди чисел первого массива, которое не входит во второй массив.
2	1. Даны массивы A и B , состоящие из n элементов. Построить массив S , каждый элемент которого равен сумме соответствующих элементов массивов A и B . 2. Определить наименьшее натуральное число, отсутствующее в последовательности из n натуральных чисел.

№ ва- рианта	Условие задачи
3	<p>1. Заданы два массива A и B. Подсчитать в них количество элементов, меньших значения t, и первым на печать вывести массив, имеющий наибольшее их количество.</p> <p>2. Даны два массива C и D, каждый из m элементов. Подсчитать количество таких чисел k, для которых: C[k] > D[k] и C[k] < D[k], C[k] = D[k].</p>
4	<p>1. Дан массив A из n элементов и B из m элементов. Содержится ли наибольший элемент массива A в массиве B?</p> <p>2. Найти количество различных чисел среди элементов целочисленного массива Z, содержащего n элементов.</p>
5	<p>1. Разделить массив на две части, поместив в первую элементы, большие среднего арифметического их суммы, а во вторую – меньшие.</p> <p>2. Даны два массива x и y. Найти количество одинаковых элементов в массивах A и B, т. е. количество пар a[i] = b[j] для некоторых i и j.</p>
6	<p>1. Дан массив C, не содержащий нулевых элементов. Преобразовать массив так, чтобы вначале шли отрицательные элементы, а затем положительные.</p> <p>2. Дан массив S, содержащий n1 элементов, массив D, содержащий n2 элементов, и число f. Найти сумму вида s[i] + d[j], наиболее близкую к числу f.</p>
7	<p>1. Задан одномерный числовой массив A из n элементов и число k. Найти номера всех элементов массива, которые равны, больше и меньше k.</p> <p>2. В целом числе N выделить цифры, кратные числу k, и записать их в одномерный массив.</p>

№ варианта	Условие задачи
8	<p>1. Дан массив X, содержащий k элементов, и массив Y, содержащий n элементов. Найти их «пересечение», т. е. массив Z, содержащий их общие элементы.</p> <p>2. Дан массив x, содержащий k элементов, массив y, содержащий n элементов, и число q. Найти сумму вида x[i] + y[j], наиболее близкую к числу q.</p>
9	<p>1. Даны N положительных целых чисел, которые не делятся ни на какие простые числа, кроме 2 и 3. Удалить из массива числа так, чтобы из любых двух оставшихся одно число делилось на другое.</p> <p>2. Определить, содержится ли наименьший элемент массива A в массиве B.</p>
10	<p>1. Заданы два массива. Найти наименьшее среди чисел первого массива, которое не входит во второй массив (считая, что хотя бы одно такое число есть).</p> <p>2. Определить массив C, каждый элемент которого равен сумме соответствующих элементов массивов A и B.</p>
11	<p>1. Вводится последовательность из n натуральных чисел. Определить наименьшее натуральное число, отсутствующее в последовательности.</p> <p>2. В двух массивах подсчитать количество элементов, меньших значения z. Вывести первым массив, имеющий наименьшее их количество.</p>
12	<p>1. Заданы два массива A и B, каждый из n элементов. Подсчитать количество таких чисел k, для которых: A[k] = B[k], A[k] > B[k] и A[k] < B[k].</p> <p>2. В одномерном массиве A найти номера всех элементов массива, которые равны, больше и меньше некоторого числа k.</p>

№ варианта	Условие задачи
13	<p>1. Дан целочисленный массив X, содержащий n элементов. Найти количество различных чисел среди элементов этого массива.</p> <p>2. Разделить массив на две части, поместив в первую элементы, меньшие среднего арифметического их суммы, а во вторую – большие.</p>
14	<p>1. Даны два массива x и y. Найти количество одинаковых элементов в этих массивах, т. е. количество пар x[i] = y[j] для некоторых i и j.</p> <p>2. Преобразовать массив K так, чтобы вначале шли числа меньшие числа t, а затем – большие.</p>
15	<p>1. Дан массив A размера n, не содержащий нулевых элементов. Преобразовать массив A так, чтобы вначале шли положительные элементы, а затем отрицательные. Дополнительные массивы не использовать.</p> <p>2. Определить, содержится ли наибольший элемент массива F в массиве D?</p>
16	<p>1. Дан массив X, содержащий k элементов, и массив Y, содержащий n элементов. Сформировать массив Z, содержащий общие элементы массивов P и Q.</p> <p>2. Массив A размера n не содержит нулевых элементов. Преобразовать массив так, чтобы вначале шли отрицательные элементы, а затем положительные.</p>

Тест "Указатели и ссылки"

[В начало практикума](#)

Лабораторная работа № 13. Обработка символьной информации

Строка символов хранится в памяти как *массив*. Каждый элемент массива содержит один символ, при этом последним символом является '\0'. Поэтому при объявлении массива надо указывать размерность на единицу больше, чем количество символов. Имя массива без индекса является **указателем-константой**.

Задание	Краткие теоретические сведения	
<p>1. Изучить способы преобразования символов, выполнив программы в правой части.</p> <p>Выполнить прокрутки программ и записать условия.</p> <p>Опробовать программы с различным текстом.</p>	<pre>#include <iostream> void main() { int num = 5; char symb, new_symbol = ' '; symb = num + '0'; std::cout<<symb<<' '; if (symb >= '0' && symb <= '9') num = symb - '0'; std::cout<<num<<' '; symb = 'b'; if (symb >= 'a' && symb <= 'z') new_symbol = symb - 'a' + 'A'; std::cout<<new_symbol<<' '; }</pre> <p>Заменить четвертую и пятую строчки на операторы:</p> <pre>char *pc; pc = &symb; *pc = num + '0'; std::cout<<*pc<<' ';</pre> <p>Объяснить результат.</p>	<pre>#include <iostream> void main() { char str[] = "Text"; int count = 0; char *pstr; pstr = &str[0]; if (str) while (*pstr++) ++count; std::cout<<count; }</pre> <p>В данной программе указатель может содержать нулевое значение, поэтому перед операцией разыменования его следует проверять.</p>

2. Выполнить программу, приведенную в правой части.

Объяснить принцип использования функций стандартной библиотеки.

Пример. Имеется адрес файла в сети, например, **http://belstu.by/p1/p2/file1.htm**. Определить имя последней папки (каталога).

```
#include <cstring>
#include <iostream>
using namespace std;

void main()
{
    char str[50], s[20];
    int i, j, i1, i2;
    cout<<"Input string:";
    cin>> str;
    for(i = strlen(str); i >= 0; i--)
        if (str[i] == '\\') break;
    i2 = i;
    for (j = (i2 - 1); j >= 0; j--)
        if (str[j] == '\\') break;
    i1 = j;
    if (i2 == i1)
        printf(" '\\' ");
    else
    { strncpy_s(s, &str[i1+1], i2-i1-1);
      s[i2 - i1 - 1] = 0;
      cout<<"katalog:"<< s<<"\n";
    }
}
```

Функ-ция	Краткое описание функции
strcmp	int strcmp(const char *str1, const char *str2); Сравнивает строки str1 и str2. Если str1 < str2, то результат <0, если str1 = str2, то результат = 0, если str1 > str2, то результат>0.
strcpy	char* strcpy(char*s1, const char *s2); Копирует байты из строки s1 в строку s2
strlen	int strlen (const char *str); Вычисляет длину строки str
strncat	char *strncat(char *s1, const char *s2, int kol); Приписывает kol символов строки s1 к строке s2
strncpy	char *strncpy(char *s1, const char *s2, int kol); Копирует kol символов строки s2 в строку s1
strnset	char *strnset(char *str, int c, int kol); Заменяет первые kol символов строки s1 символом c
atoi	int atoi(char *str); Преобразует числовые символы в целое число
atof	float atof(char *str); Преобразует числовые символы в вещественное число

<p>3. Изучить способы работы с символьными массивами, выполнив программу в правой части.</p>	<pre> void main() { char s, t[20]; int i, j, n; cout<<"Input string: "; gets_s(t); cout<<"Input symbol: "; cin>>s; n = strlen(t); for(i = 0; i <= n; i++) if (t[i] == s) for(j = i; j <= n; j++) t[i] = t[i + 1]; for(i = 0; i < n; i++) cout << t[i]; } </pre>	<p><u>Пример.</u> Из предложения удалить все символы, совпадающие с символом, введенным с клавиатуры.</p>
<p>4. Выполнить программу в правой части.</p> <p>Закомментировать строку cin.ignore...</p> <p>Объяснить возникшие проблемы при выполнении программы.</p>	<pre> void main() { char t[3][20]; int i, n; cout<<"Input size (<=3): "; cin>>n; cin.ignore(cin.rdbuf()->in_avail()); for(i = 0; i < n; i++) { cout<<"Input string: "; gets_s(t[i]); } for(i = 0; i < n; i++) puts(t[i]); } </pre>	<p>У потока ввода есть буфер чтения, в котором он хранит символы. При неоднократном чтении символов в буфере могут остаться ненужные символы.</p> <p>Буфер очищается с помощью:</p> <p>cin.ignore(cin.rdbuf()->in_avail());</p> <p>Функция rdbuf() обращается к буферу, функция in_avail() определяет количество символов, которое нужно проигнорировать с использованием ignore().</p>

5. В правой части записаны два варианта решения задачи. Выполнить программы и объяснить различия между ними.

Внести изменения в программу с тем, чтобы проверялось не только количество скобок, но и правильность их расстановки (первой в тексте должна быть открывающая скобка).

Пример проверки соответствия количества открывающих и закрывающих круглых скобок в строке.

```
#include <stdio.h>
void main()
{
    char s[256];
    int i, count;
    puts("Enter string: ");
    gets_s(s);
    for(count=i=0; s[i] != 0; i++)
    {
        if (s[i] == '(') count++;
        if (s[i] == ')') count--;
    }
    if (!count)
        puts("Ok\n");
    else
        puts("Not Ok\n");
}
```

```
#include <stdio.h>
void main()
{
    char s[256];
    int count;
    char *ps;
    puts("Enter string: ");
    gets_s(s);
    for(count = 0, ps = s; *ps != 0; ps++)
    {
        if (*ps == '(') count++;
        if (*ps == ')') count--;
    }
    if (count == 0)
        puts("Ok\n");
    else
        puts("Not Ok\n");
}
```

6. Выполнить задания из таблицы ниже, используя **индексы** для доступа к элементам массивов в первой программе и **указатели** – во второй программе.

При написании программ не использовать стандартные функции для строк символов.

№ варианта	Условие задачи
1	<p>1. Написать программу, реализующую вставку в строку n символов, начиная с позиции k.</p> <p>2. Дана строка, состоящая из групп нулей и единиц. Каждая группа отделяется от другой одним или несколькими пробелами. Найти количество групп с пятью символами.</p>
2	<p>1. Написать программу, реализующую выделение подстроки S1 длиной k с позиции номер n из некоторой строки.</p> <p>2. Дана строка, состоящая из групп нулей и единиц. Определить самую короткую группу.</p>
3	<p>1. В строке есть два символа *. Получить все символы между первым и вторым символом *.</p> <p>2. Дана строка, состоящая из букв, цифр, запятых, точек, знаков + и –. Выделить подстроку, которая соответствует записи целого числа.</p>
4	<p>1. Написать программу, которая удаляет в строке все буквы b в тексте, написанном латинскими буквами.</p> <p>2. Дана строка символов, состоящая из произвольных десятичных цифр, разделенных пробелами. Найти числа этой строки в порядке возрастания их значений.</p>
5	<p>1. Исключить из строки группы символов, расположенные между скобками вместе со скобками. Предполагается, что нет вложенных скобок.</p> <p>2. Дана строка символов, состоящая из произвольного текста на английском языке, слова разделены пробелами. Определить слова этого текста в порядке, соответствующем латинскому алфавиту.</p>
6	<p>1. В строке есть символы *. Преобразовать строку следующим образом: удалить все символы *, и повторить каждый символ, отличный от *.</p>

№ варианта	Условие задачи
	2. Дана строка символов, состоящая из произвольного текста, слова разделены пробелами. Найти порядковый номер слова минимальной длины и количество символов в этом слове.
7	<p>1. Преобразовать строку: после каждой буквы a добавить символ «!».</p> <p>2. Дана строка символов, состоящая из букв, цифр, запятых, точек, знаков «+» и «-». Найти количество цифр.</p>
8	<p>1. Написать программу, которая осуществляет сравнение двух строк и выводит сообщение о том, какие символы совпадают.</p> <p>2. Дана строка, состоящая из групп нулей и единиц. Найти группы с четным количеством символов.</p>
9	<p>1. Написать программу, реализующую вставку подстроки St длиной n1 в строку S с позиции номер n2.</p> <p>2. Дана строка, состоящая из групп нулей и единиц. Подсчитать количество символов в самой длинной группе.</p>
10	<p>1. Написать программу, которая записывает строку в обратном порядке.</p> <p>2. Дана строка символов, состоящая из произвольного текста, слова разделены пробелами. Разбить исходную строку на две подстроки. Первая подстрока должна иметь длину k символов (если на k-ю позицию попадает слово, то его следует отнести ко второй строке).</p>
11	<p>1. Вывести текст, составленный из последних букв всех слов.</p> <p>2. Дана строка символов, состоящая из произвольного текста, слова разделены пробелами. Найти порядковый номер слова максимальной длины и номер позиции в строке, с которой оно начинается.</p>

№ варианта	Условие задачи
12	<p>1. Зашифровать введенную с клавиатуры строку, поменяв местами первый символ со вторым, третий с четвертым и т. д.</p> <p>2. Дана строка символов, состоящая из букв, цифр, запятых, точек, знаков «+» и «-». Найти количество запятых и точек.</p>
13	<p>1. Отредактировать заданное предложение, удаляя из него все слова с чётными номерами.</p> <p>2. Дана строка символов, состоящая из произвольного текста, слова разделены пробелами. Определить порядковый номер слова, один из символов которого находится на k-й позиции (если на k-ю позицию попадает пробел, то найти номер предыдущего слова).</p>
14	<p>1. Найти самое длинное и самое короткое слово в заданном предложении.</p> <p>2. Дана строка символов, состоящая из произвольных десятичных цифр, разделенных пробелами. Вывести четные числа этой строки.</p>
15	<p>1. Из текста удалить те его части, которые заключены в кавычки (вместе с кавычками).</p> <p>2. Дана строка, состоящая из групп нулей и единиц. Подсчитать количество единиц в группах с нечетным количеством символов.</p>
16	<p>1. Из предложения удалить все символы, совпадающие с символом +.</p> <p>2. Дана строка символов, состоящая из произвольных десятичных цифр, разделенных пробелами. Найти числа этой строки в порядке убывания их значений.</p>

5. Дополнительные задания.

1. В заданной последовательности слов найти все слова, имеющие заданное окончание.
2. В имеющемся словаре найти группы слов, записанных одними и теми же буквами и отличающиеся только порядком расположения.
3. Из заданного предложения удалить те слова, которые уже встречались в предложении раньше.
4. Отредактировать заданное предложение, удаляя из него все слова с нечетными номерами и переворачивая слова с четными номерами. Пример: HOW DO YOU DO преобразовать в OD OD.
5. Даны два предложения. Найти самое короткое из слов первого предложения, которого нет во втором предложении.

Тест "Обработка символьной информации"

Контрольная работа¹ 2

[В начало практикума](#)

Лабораторная работа № 14. Многомерные массивы

Многомерными называются массивы, имеющие два и более индексов, которые заключены в квадратные скобки.

Имя массива без индекса является **указателем-константой** на массив из указателей, каждый из которых содержит адреса начальных элементов строк. Для доступа к элементу `A[1][2]` двумерного массива через указатели можно использовать следующую форму записи: `*(A + 1) + 2` или `*(A[1] + 2)` или `*(A + 1)[2]`.

Задание	Краткие теоретические сведения	
<p>1. Выполнить программу, записанную в правой части. Внести изменения в программу с тем, чтобы инициализировался другой массив. Осуществить вывод этого массива в виде матрицы.</p>	<p><u>Пример</u> программы, которая инициализирует массив и выводит его элементы на экран.</p> <pre>#include <iostream> void main() { const int ci = 3, const int cj = 2; int a[ci][cj] = { { 1, 2 }, { 3, 4 }, { 5, 6 } }; for (int i = 0; i < ci; i++) for (int j = 0; j < cj; j++) std::cout << "\n a[" << i << ", " << j << "] = " << a[i][j]; }</pre>	
<p>2. Выполнить программы, записанные в правой части для одной и той же задачи, условие которой надо определить. Одна из программ использует указатели.</p>	<pre>#include <iostream> void main() { const int n = 3, m = 2, h = 0; int A[n][m] = { 0, 1, 2, 3, 4, 5 }; for(int i = 0; i < n; i++) for(int j = 0; j < m; j++) h += A[i][j]; std::cout << h; }</pre>	<pre>#include <iostream> void main() { const int n = 3, m = 2, h = 0; int C[n][m] = { 0, 1, 2, 3, 4, 5 }; for(int i = 0; i < n; i++) for(int j = 0; j < m; j++) h += (*(C + i) + j); std::cout << h; }</pre>

3. Изучить способы работы с двумерными массивами, выполнив программу в правой части и записав ее условие.

Внести изменения в программу с тем, чтобы определялись минимальные элементы каждой строки. Для доступа к элементам матрицы использовать указатели.

```
#include <iostream>
void main()
{
    setlocale(LC_CTYPE, "Russian");
    using namespace std;
    const int n = 2, m = 4;
    int B[n][m];
    int i, j, row = 0, column = 0;
    cout << "Введите элементы массива" << endl;
    for (i = 0; i < n; i++)
        for (j = 0; j < m; j++)
            cin >> B[i][j];
    int min = B[0][0];
    for (int i = 0; i < n; i++)
        for (int j = 0; j < m; j++)
            if (min > B[i][j])
            {
                min = B[i][j];
                column = i;
                row = j;
            }
    cout << " Исходный массив:" << endl;
    for (int i = 0; i < n; i++)
    {
        cout << "\n";
        for (int j = 0; j < m; j++)
            cout << "B[" << i << ", " << j << "] =" << B[i][j] << "\t";
    }
    cout << endl;
    cout << "Минимальный элемент B[" << column << ", " << row << "] =" << min << endl;
}
```

4. Выполнить задания из таблицы ниже, используя *индексы* для доступа к элементам массивов в первой программе и *указатели* – во второй программе.

№ варианта	Условие задачи
1	<p>1. Найти наибольший элемент матрицы $A(n, m)$, а также номера строки и столбца, на пересечении которых он находится.</p> <p>2. Задана матрица размером $n \times m$ и число k. В переменную b поместить значение 0, если все элементы k-го столбца матрицы нулевые, и значение 1 – в противном случае.</p>
2	<p>1. В каждой строке заданной матрицы $A(n, m)$ вычислить сумму, количество и среднее арифметическое положительных элементов.</p> <p>2. Определить количество различных элементов матрицы размером $n \times m$ (повторяющиеся элементы считать один раз).</p>
3	<p>1. Для заданной целочисленной матрицы $A(n, m)$ определить, является ли сумма её элементов чётным числом.</p> <p>2. Задана матрица размером $n \times m$ и число k. В переменную f поместить значение 0, если все элементы k-ой строки упорядочены по убыванию, и значение 1 – в противном случае.</p>
4	<p>1. Дана матрица $A(n, m)$. Найти количество элементов этой матрицы, больших среднего арифметического всех её элементов.</p> <p>2. Задана числовая матрица размером $n \times m$. Определить количество таких элементов матрицы, у которых в строке слева от элемента находятся числа, меньшие его, а справа – большие.</p>

№ варианта	Условие задачи
5	<p>1. Дана целочисленная матрица $A(n, m)$. Вычислить сумму и произведение тех её элементов, которые при делении на два дают нечётное число.</p> <p>2. Проверить, есть ли в матрице хотя бы одна строка, содержащая положительный элемент, и найти ее номер. Знаки элементов предыдущей строки изменить на противоположные.</p>
6	<p>1. В заданной матрице $A(n, m)$ поменять местами столбцы с номерами p и q.</p> <p>2. Найти в матрице первую строку, все элементы которой положительны, и сумму этих элементов. Уменьшить все элементы матрицы на эту сумму.</p>
7	<p>1. Дана матрица $A(n, m)$. Поменять местами её наибольший и наименьший элементы.</p> <p>2. Найти в матрице первую строку, все элементы которой равны нулю. Все элементы столбца с таким же номером уменьшить вдвое.</p>
8	<p>1. Даны две целочисленные матрицы $A(n, m)$ и $B(n, m)$. Подсчитать количество тех пар (a_{ij}, b_{ij}), для которых: а) $a_{ij} < b_{ij}$; б) $a_{ij} = b_{ij}$; в) $a_{ij} > b_{ij}$.</p> <p>2. Проверить, есть ли в матрице хотя бы одна строка, содержащая отрицательный элемент, и найти ее номер. Все элементы столбца с таким же номером уменьшить вдвое.</p>
9	<p>1. Дана матрица $A(n, n)$. Переписать элементы её главной диагонали в одномерный массив $Y(N)$ и разделить их на максимальный элемент главной диагонали.</p> <p>2. Найти в матрице первый столбец, все элементы которого положительны. Знаки элементов предыдущего столбца изменить на противоположные.</p>

№ варианта	Условие задачи
10	<p>1. Дана матрица B(n, m). Вычислить произведение чётных положительных элементов матрицы.</p> <p>2. Проверить, есть ли в матрице хотя бы одна строка, содержащая элемент, равный нулю, и найти ее номер. Уменьшить все элементы матрицы на значение первого элемента найденной строки.</p>
11	<p>1. Найти наибольший элемент главной диагонали матрицы A(n, n) и вывести на печать всю строку, в которой он находится.</p> <p>2. Определить в матрице первый столбец, все элементы которого отрицательны, и среднее арифметическое этих элементов. Вычесть полученное значение из всех элементов матрицы.</p>
12	<p>1. Дана целочисленная матрица A(n, m). Вычислить сумму и произведение нечётных отрицательных элементов матрицы, удовлетворяющих условию $a_{ij} < i$.</p> <p>2. Проверить, все ли строки матрицы содержат хотя бы один положительный элемент. Если да, то изменить знаки всех элементов матрицы на обратные.</p>
13	<p>1. Найти наименьший элемент главной диагонали матрицы C(n, n) и вывести на печать столбец, в котором он находится.</p> <p>2. Задана числовая матрица размером n x m. Определить количество таких элементов матрицы, значение каждого из которых больше суммы остальных элементов своего столбца.</p>
14	<p>1. Дана матрица A(n, n) и целое число m. Преобразовать матрицу по правилу: строку с номером m сделать столбцом с номером m, а столбец с номером m сделать строкой с номером m.</p> <p>2. Найти в матрице первую строку, все элементы которой равны нулю. Все элементы столбца с таким же номером уменьшить вдвое.</p>

№ варианта	Условие задачи
15	<p>1. В заданном массиве $A(n, n)$ вычислить две суммы элементов, расположенных выше и ниже главной диагонали.</p> <p>2. Найти в матрице первый столбец, все элементы которого равны нулю. Знаки элементов строки с таким же номером изменить на противоположные.</p>
16	<p>1. Найти элемент, наиболее близкий к среднему арифметическому всех элементов матрицы $B(n, n)$, а также номера строки и столбца, на пересечении которых он находится.</p> <p>2. Проверить, все ли строки матрицы содержат хотя бы один нулевой элемент. Если нет, то заменить значения всех отрицательных элементов матрицы на нулевые.</p>

6. Дополнительные задания.

1. Дана квадратная матрица порядка $2n$, элементы которой формируются случайным образом и находятся в пределах от -10 до 10 . Получить новую матрицу, переставляя ее блоки размера $n \times n$ в соответствии со схемой.



2. Латинским квадратом порядка n называется квадратная таблица размером $n \times n$, каждая строка и каждый столбец которой содержат все числа от 1 до n . Для заданного n в матрице $L(n, n)$ построить латинский квадрат порядка n .

3. Путем перестановки элементов квадратной вещественной матрицы добиться того, чтобы ее максимальный элемент находился в левом верхнем углу, следующий по величине – в позиции (2, 2), следующий – в позиции (3, 3) и т. д., заполнив таким образом всю главную диагональ.

Тест "Многомерные массивы"

[В начало практикума](#)

Лабораторная работа № 15. Динамическое выделение памяти

Динамическая память – это область памяти, которая выделяется во время *выполнения* программы. После использования памяти ее надо освобождать, что позволит эффективно расходовать память.

Формирование динамических массивов можно организовать двумя способами: с использованием функций (язык C); с использованием операторов **new** и **delete** (язык C++).

Задание	Краткие теоретические сведения	
<p>1. Изучить формирование динамического массива с использованием стандартных функций, выполнив программу на языке C, записанную справа.</p> <p>Написать пояснения к программе.</p> <p>Добавить операторы вычисления максимального по модулю элемента массива.</p>	Ниже представлены функции работы с динамической памятью и программа с их использованием.	
	Функция	Прототип и краткое описание
	malloc	void *malloc(unsigned s) – возвращает указатель на начало области динамической памяти длиной в s байт, при неудачном завершении возвращает NULL
	calloc	void *calloc(unsigned n, unsigned m) – возвращает указатель на начало динамической памяти для размещения n элементов длиной по m байт, при неудачном завершении возвращает NULL
	realloc	void *realloc(void *p, unsigned s) – изменяет размер блока ранее выделенной динамической памяти до размера s байт, p – адрес начала изменяемого блока, при неудачном завершении возвращает NULL
	free	void *free(void p) – освобождает ранее выделенный участок динамической памяти, p – адрес первого байта

Пример. Программа выделяет динамическую память под массив из **n** элементов, вводит массив с клавиатуры и выводит его на экран.

```
#include <stdio.h>
#include <stdlib.h>

void main()
{
    int *ptr, i, n;
    printf("Input size of massiv, n<30 \n");
    scanf("%d", &n);

    if(!(ptr = (int*)malloc(n*sizeof(int)))) //выделение памяти и проверка,
    {                                         //достаточно ли для нее места
        puts("Not enough memory");
        return;
    }
    for (i = 0; i < n; i++)
    {
        printf("Input element [%d]\n" , i + 1);
        scanf("%d", ptr + i);
    }
    printf("\nMassiv: \n", i + 1);
    for (i = 0; i < n; i++)
        printf("%d ", *(ptr + i));

    free(ptr); //освобождение динамической памяти
}
```

2. Изучить способы выделения динамической памяти для *одномерного массива*, выполнив программу на языке *C++*, записанную в правой части.

Опробовать работу программы с разными значениями вводимых символов.

Операция **new** позволяет выделить и сделать доступным свободный участок памяти, размеры которого соответствуют типу данных. Оператор возвращает указатель на начало выделенного блока памяти. Операция **delete** освобождает участок памяти, ранее выделенный операцией **new**. Аргументом оператора **delete** выступает адрес первой ячейки блока, который необходимо освободить.

```
#include <iostream>
using namespace std;
void main()
{
    setlocale(LC_CTYPE, "Russian");
    char *pStr;    // указатель для адреса массива
    char c, *pc;
    int j, length, count = 0;
    cout << "Введите размер строки ";
    cin >> length;
    pStr = new char[length + 1]; //выделение динам.памяти
    cout << "Введите строку ";
    cin >> pStr;
    cout << "Введите символ ";
    cin >> c;
    pc = &c;
    for (j = 0; j < length; j++)
        if (pStr[j] == *pc)
            count++;
    cout << count << " повторений символа " << c << endl;
    delete[] pStr; // освобождение динам. памяти
}
```

Пример. Пусть необходимо в строке, введенной с клавиатуры, подсчитать количество повторений символа, который также вводится с клавиатуры.

В программе при выделении динамической памяти количество элементов символьного массива *s* увеличено на 1, так как в конце массива должен присутствовать признак конца строки (нуль-байт).

3. В программе, записанной справа, демонстрируется использование динамической памяти при работе с *двумерным массивом*.

Выполнить программу с различными размерами исходного массива.

```
#include <ctime>
#include <iostream>
using namespace std;
void main()
{   int size, mult = 1, sum = 0, **A;
    cout << "Enter size ";
    cin >> size;
    srand(time(0));
    A = new int*[size];    //выделение динам.памяти
    for (int i = 0; i < size; i++)
        A[i] = new int[size];
    for (int j = 0; j < size; j++)
        A[i][j] = 1 + rand() % 10;
    for (int **ptr = A; ptr != A + size; ++ptr)
    {   for (int* it = *ptr; it != *ptr + size; ++it)
        {   cout << "\t" << *it;
            if (*it % 2 == 0)
            { mult *= *it;
              sum += *it;
            }
        }
        cout << endl;
    }
    cout << "sum: " << sum << endl;
    cout << "mult: " << mult << endl;
    for(int k=0; k < size; k++) //освобождение памяти
        delete[] A[k];
    delete[] A;
}
```

Пример. Дана целочисленная матрица случайных чисел размерности $n \times m$.

Вычислить сумму и произведение ее четных элементов.

4. В соответствии со своим вариантом написать программы с использованием *динамических* массивов для условий задач из таблицы. Начальные размерности массивов ввести с клавиатуры. Первое задание выполнить с использованием *функций* для выделения динамической памяти на языке C, второе – с использованием операторов **new** и **delete** на языке C++.

№ варианта	Условия задач
1	<p>1. В одномерном массиве, состоящем из n вещественных элементов, вычислить количество отрицательных элементов массива и сумму модулей элементов, расположенных после минимального по модулю элемента.</p> <p>2. Найти в матрице первый столбец, все элементы которого положительны. Знаки элементов предыдущего столбца изменить на противоположные.</p>
2	<p>1. Задан массив A из n элементов. Проверить, есть ли в нём отрицательные элементы. Если есть, то найти наибольшее значение k, при котором $A[k] < 0$.</p> <p>2. Дана матрица. Переставляя ее строки и столбцы, добиться того, чтобы наибольший элемент оказался в верхнем левом углу.</p>
3	<p>1. В одномерном массиве, состоящем из n вещественных элементов, вычислить произведение отрицательных элементов массива и сумму положительных элементов массива, расположенных до максимального элемента.</p> <p>2. Дана матрица размером 4×4. Найти сумму наименьших элементов ее нечетных строк и наибольших элементов ее четных строк.</p>
4	<p>1. В одномерном массиве, состоящем из n вещественных элементов, вычислить номер минимального элемента массива и сумму элементов массива, расположенных между первым и вторым отрицательными элементами.</p> <p>2. Дана целочисленная прямоугольная матрица. Определить количество столбцов, не содержащих ни одного нулевого элемента.</p>

№ варианта	Условия задач
5	<p>1. В одномерном массиве, состоящем из n вещественных элементов, вычислить количество элементов массива, равных 0, и сумму элементов массива, расположенных после минимального элемента.</p> <p>2. Дана целочисленная прямоугольная матрица. Определить количество столбцов, содержащих хотя бы один нулевой элемент и номер строки, в которой находится самая длинная серия одинаковых элементов.</p>
6	<p>1. В одномерном массиве, состоящем из n вещественных элементов, вычислить минимальный элемент массива и сумму элементов, расположенных между первым и последним положительными элементами.</p> <p>2. Проверить, есть ли в матрице хотя бы одна строка, содержащая отрицательный элемент, и найти ее номер. Все элементы столбца с таким же номером уменьшить вдвое.</p>
7	<p>1. Задан массив A из n элементов. Найти количество элементов этого массива, больших среднего арифметического всех его элементов.</p> <p>2. Дана целочисленная прямоугольная матрица. Определить количество строк, не содержащих ни одного нулевого элемента и максимальное из чисел, встречающихся в заданной матрице более одного раза.</p>
8	<p>1. В одномерном массиве, состоящем из k целых элементов, вычислить количество положительных элементов массива и сумму элементов массива, расположенных после последнего элемента, равного нулю.</p> <p>2. Дана целочисленная прямоугольная матрица. Определить номер первого из столбцов, содержащих хотя бы один нулевой элемент.</p>
9	<p>1. В одномерном массиве, состоящем из n вещественных элементов, вычислить произведение положительных элементов массива и сумму элементов массива, расположенных до минимального элемента.</p> <p>2. Найти в матрице первую строку, все элементы которой положительны, и сумму этих элементов. Уменьшить все элементы матрицы на эту сумму.</p>

№ варианта	Условия задач
10	<p>1. В одномерном массиве, состоящем из n вещественных элементов, вычислить сумму положительных элементов массива и произведение элементов массива, расположенных между максимальным по модулю и минимальным по модулю элементами.</p> <p>2. Найти в матрице первую строку, все элементы которой отрицательны. Увеличить все элементы матрицы на значение первого элемента найденной строки.</p>
11	<p>1. Задан массив A из n элементов. Проверить, есть ли в нём элементы, равные нулю. Если есть, найти наименьшее k, при котором $A[k] = 0$.</p> <p>2. Для заданной матрицы размером 4 на 4 найти такие k, при которых k-я строка матрицы совпадает с k-м столбцом.</p>
12	<p>1. Задан массив A из n элементов. Подсчитать, сколько раз встречается в нем максимальное число.</p> <p>2. Проверить, есть ли в матрице хотя бы одна строка, содержащая положительный элемент, и найти ее номер. Знаки элементов предыдущей строки изменить на противоположные.</p>
13	<p>1. В одномерном массиве, состоящем из вещественных элементов, вычислить сумму элементов массива с нечетными номерами и сумму элементов массива, расположенных между первым и последним отрицательными элементами.</p> <p>2. Найти в матрице первую строку, все элементы которой равны нулю. Все элементы столбца с таким же номером уменьшить вдвое.</p>
14	<p>1. В массиве, состоящем из вещественных элементов, вычислить количество элементов массива, больших некоторого заданного числа C, и произведение элементов массива, расположенных после максимального по модулю элемента.</p>

№ ва- рианта	Условия задач
	<p>2. Проверить, есть ли в матрице хотя бы одна строка, содержащая элемент, равный нулю, и найти ее номер. Уменьшить все элементы матрицы на значение первого элемента найденной строки.</p>
15	<p>1. В одномерном массиве, состоящем из n вещественных элементов, вычислить количество элементов массива, меньших некоторого заданного числа c, и сумму целых частей элементов массива, расположенных после последнего отрицательного элемента.</p> <p>2. Проверить, все ли строки матрицы содержат хотя бы один нулевой элемент. Если нет, то заменить значения всех отрицательных элементов матрицы на нулевые.</p>
16	<p>1. В одномерном массиве, состоящем из вещественных элементов, вычислить номер минимального по модулю элемента массива и сумму модулей элементов массива, расположенных после первого отрицательного элемента.</p> <p>2. Дана целочисленная квадратная матрица. Определить произведение элементов в тех строках, которые не содержат отрицательных элементов и максимум среди сумм элементов диагоналей, параллельных главной диагонали матрицы.</p>

Тест "Динамические массивы"

Контрольная работа ¹ 3

[В начало практикума](#)

Лабораторная работа № 16. Функции пользователя

В любой программе C/C++ должна быть функция с именем **main** (главная функция или основная программа) с которой начинается решение задачи. Могут также присутствовать *функции пользователя* (подпрограммы). Функция пользователя может быть записана как до основной программы, так и после. В последнем случае вызов функции следует предварять объявлением (прототипом) этой функции.

При вызове функции значения фактических параметров из вызывающей программы пересылаются в формальные параметры, записанные в определении функции пользователя.

Задание	Краткие теоретические сведения		
1. Изучить способы передачи значений <i>переменных</i> в функцию, выполнив программы, записанные в правой части. Записать условие задачи.	Передача параметра по значению	Передача параметра по указателю	Передача параметра по ссылке
	<pre>int square(int); void main() { for(int x=1; x<=10; x++) cout<<square(x)<< ' '; } int square(int y) { return y * y; }</pre>	<pre>int square(int *); void main() { for(int x=1; x<=10; x++) cout<<square(&x)<< ' '; } int square(int* y) { return *y * *y; }</pre>	<pre>int square(int &); void main() { for(int x=1; x<=10; x++) cout<<square(x)<< ' '; } int square(int& y) { return y * y; }</pre>
	<i>Значение</i> фактического параметра x копируется в локальную переменную y , являющуюся формальным параметром.	<i>Адрес</i> фактического параметра x передается в указатель y . Для получения значения в функции пользователя требуется операция разыменования.	Параметр y выступает <i>псевдонимом</i> фактического параметра x , то есть передаётся сам объект. Нет необходимости в разыменовании указателя.

2. Изучить использование указателей и ссылок как формальных параметров функции при работе с одномерным массивом данных, опробовав работу программ в правой части.

Написать условие задачи.

Если в качестве передаваемого аргумента используется *массив* данных, то возможна передача только по *указателю* или по *ссылке*.

Имя одномерного массива без индекса является **адресом** элемента с **нулевым** индексом. При передаче в функцию имя преобразуется в *указатель*, и копия указателя на начало массива передается в функцию. Можно использовать несколько вариантов передачи массива в функцию пользователя.

1. В примере **mas** и **x** – имена массивов, то есть являются указателями.

```
int sum(int x[])// определение функции
{
    int res = 0;
    for(int i = 0; i < 5; i++)
        res += x[i];
    return res;
}
int main()
{
    int mas[5], i;
    for(i = 0; i < 5; i++)
    {
        mas[i] = i * i;
        cout<< mas[i] <<' ';
    }
    cout <<endl<< "result=" << sum(mas) << endl;
}
```

2. В другом варианте определения функции *указатель* используется явно:

```
int sum(int *x)
```

Текст функции пользователя и ее вызов прежние.

3. При передаче массива по *ссылке* можно изменять значение указателя на массив.

	<pre> int sum(int &x) { int *p = &x, res = 0; for(int i = 0; i < 5; i++) res += p[i]; return res; } </pre> <p>Вызов функции: <code>cout<<endl <<"result="<<sum(*mas)<<endl;</code></p>
<p>3. В программе, записанной справа, используется меню, разработанное с помощью оператора switch, которое позволяет делать выбор между двумя функциями.</p> <p>Написать условие задачи и комментарии к программе.</p>	<pre> #include <iostream> using namespace std; int massiv(int *M, int); //прототип void matrix(); //прототип int main(void) { setlocale(LC_CTYPE, "Russian"); int choice; do { cout << "Выберите вариант работы" << endl; cout << "1 - с одномерным массивом" << endl; cout << "2 - с матрицей" << endl; cout << "3 - выход" << endl; cin >> choice; switch (choice) { </pre>

```

        case 1: int *M, size, i, av;
                cout << "Введите размер массива ";
                cin >> size;
                M = new int[size];
                for (i = 0; i < size; i++)
                {   cout << "Введите " << i + 1 << " число ";
                    cin >> *(M + i);
                }
                av = massiv(M, size);
                cout << "Результат=" << av << endl;
                delete[] M;
                break;
        case 2: matrix();
                break;
        case 3: break;
    }
} while (c != 3);
}

int massiv(int *M, int size)
{   int sum = 0, avar;
    for (int i = 0; i < size; i++)
        sum += *(M + i);
    avar = sum / size;
    return avar;
}

void matrix()
{   int **A, row, col, i, j, m;
    cout << "Введите число строк матрицы ";
    cin >> row;

```

```

    cout << "Введите число столбцов ";
    cin >> col;
    A = new int*[row];
    for (i = 0; i < row; i++)
        A[i] = new int[col];
    for (i = 0; i < row; i++)
        for (j = 0; j < col; j++)
        {
            cout << "Введите A[" << i << "],[" << j << "]= ";
            cin >> (*(A + i) + j);
        }
    m = A[0][0];
    for (i = 0; i < row; i++)
        for (j = 0; j < col; j++)
            if (*(A + i) + j > m)
                m = (*(A + i) + j);
    cout << "Результат=" << m << endl;
    for (i = 0; i < row; i++)
        delete A[i];
    delete[] A;
}

```

4. В соответствии со своим вариантом написать программы с использованием *динамических* массивов и *функций пользователя* для условий задач из таблицы.

Объединить написанные программы, разработав интерфейс с помощью оператора **switch**. Начальные размерности и значения элементов массивов ввести с клавиатуры в главной функции.

Массивы передать из главной функции в функции пользователя как *параметры*.

№ варианта	Условия задач
1	<p>1. Ввести целые числа в массив $A[n, m]$ ($n, m > 0$) и все отрицательные элементы в нечетных строках заменить на такие же положительные.</p> <p>2. Написать программу, преобразующую строку, состоящую только из прописных букв, в строку, состоящую из прописных и строчных букв, при этом первая буква после точки и пробела – прописная, остальные – строчные.</p>
2	<p>1. Ввести вещественные числа в массив $A[n, m]$ ($n, m > 0$) и ко всем числам каждой строки прибавить единицу. Если первый элемент строки нулевой, ничего не делать.</p> <p>2. Дана строка. Все русские буквы привести к верхнему регистру, латинские – заменить символом «?». Вывести результат на экран.</p>
3	<p>1. Найти сумму элементов, лежащих ниже главной диагонали в целочисленном массиве $A[n, n]$.</p> <p>2. Написать программу, которая вводит несколько строк текста и символ «m» и использует функцию, чтобы определить суммарное число вхождений символа в текст.</p>
4	<p>1. Ввести целочисленный массив $A[n, n]$ и вычесть из каждой строки предыдущую. Из первой строки вычесть последнюю строку. Получившийся массив вывести.</p> <p>2. Дана строка. Слова в предложении разделены одним или несколькими пробелами. Слова могут состоять только из цифр или букв. Найти сумму чисел, входящих в строку.</p>
5	<p>1. Дан двумерный массив. В каждой строке выбрать минимальный элемент и среди них максимальный. Напечатать номер строки, в которой расположен этот элемент.</p> <p>2. Написать программу, которая вводит несколько строк текста и символ поиска и определяет суммарное число вхождений символа в текст.</p>

№ варианта	Условия задач
6	<p>1. Изменить массив A[n, n] так, чтобы в строках остались числа, которые встречаются более одного раза, остальные заменить нулём.</p> <p>2. Определить количество слов в строке, которые слева и справа читаются одинаково (палиндромы). Слова разделены пробелами.</p>
7	<p>1. Дан двумерный массив, состоящий из N строк и M столбцов, а также число d. Найти строку, содержащую число d.</p> <p>2. В строке подсчитать сумму кодов символов каждого слова. Слова в строке разделены пробелами.</p>
8	<p>1. Дан двумерный массив, состоящий из N строк и M столбцов. Найти количество отрицательных, положительных и нулевых элементов массива.</p> <p>2. Написать программу, которая вводит несколько слов текста и определяет общее количество слов. Слова разделены пробелами.</p>
9	<p>1. Дан двумерный массив, состоящий из N строк и M столбцов, а также число number. Проверить, находится ли это число на главной диагонали.</p> <p>2. Написать программу, которая вводит ряд строк текста и выводит те из них, которые начинаются с буквы «b».</p>
10	<p>1. Дан двумерный массив, состоящий из N строк и M столбцов, а также число k. Найти столбец, содержащий это число.</p> <p>2. Дана строка слов, разделенных пробелами, в конце строки точка. Поменять местами два центральных слова, если их количество четно.</p>

№ варианта	Условия задач
11	<p>1. Дан массив размерностью $N \times M$. Найти максимальный элемент и его позицию. Если максимальных элементов несколько, то вывести их все. Найти сумму элементов ниже главной диагонали.</p> <p>2. Преобразовать строку: после каждой буквы с добавить символ *.</p>
12	<p>1. В каждой строке матрицы $F2(d, p)$, $d \leq 16$, $p \leq 18$ заменить каждый элемент, стоящий на главной диагонали, минимальным элементом строки.</p> <p>2. Для заданной матрицы размером 4 на 4 найти такие k, при которых k-я строка матрицы совпадает с k-м столбцом.</p>
13	<p>1. В матрице $A[k, n]$, $k \leq 12$, $n \leq 8$ поменять местами строку, содержащую элемент a со строкой, содержащей элемент b.</p> <p>2. Дано предложение. Составить программу, которая выводит все слова, оканчивающиеся на -ая.</p>
14	<p>1. В одномерном массиве, состоящем из вещественных элементов, вычислить номер минимального по модулю элемента массива и сумму модулей элементов массива, расположенных после первого отрицательного элемента.</p> <p>2. Написать программу, определяющую есть ли во введенном с клавиатуры тексте слово "БГТУ".</p>
15	<p>1. В массиве, состоящем из вещественных элементов, вычислить количество элементов массива, больших некоторого заданного числа C, и произведение элементов массива, расположенных после максимального по модулю элемента.</p> <p>2. Написать программу, преобразующую строку, состоящую из строчных букв, в строку, состоящую из заглавных букв до первого пробела.</p>

№ варианта	Условия задач
16	<p>1. Задан массив A из n чисел. Подсчитать, сколько раз встречается в нем максимальное число.</p> <p>2. Определить самое короткое и самое длинное слово в строке.</p>

Тест "Функции пользователя"

[В начало практикума](#)

Лабораторная работа № 17. Указатели и ссылки при работе с функциями

Функция пользователя может принимать *параметры* в виде указателей и ссылок, а также возвращать указатели и ссылки в качестве *результата*.

Задание	Краткие теоретические сведения	
<p>1. Выполнив программу, записанную в правой части, изучить один из способов передачи <i>динамической матрицы</i> в функцию пользователя.</p> <p>Записать условие задачи.</p>	<pre>#include <iostream> using namespace std; int sum(int **, int, int); int main() { int i, j; int **matr = new int*[5]; for (i = 0; i < 5; i++) matr[i] = new int[4]; for(i = 0; i < 5; i++) { for(j = 0; j < 4; j++) { matr[i][j] = i + j; cout<< matr[i][j] <<' '; } cout <<endl; } cout<<"S="<< sum(matr, 5, 4) << endl; for(int i = 0; i < 5; i++) delete matr[i]; delete [] matr; }</pre>	<pre>int sum(int **x, int n, int m) { int res = 0; for(int i = 0; i < n; i++) for(int j = 0; j < m; j++) res += x[i][j]; return res; }</pre>

2. Выполнив программу, записанную в правой части, изучить один из способов передачи *одномерного массива* в функцию, а также использование указателя как *результата* выполнения функции.

Указатель-результат функции может указывать и на отдельную переменную, и на массив.

```
#include <iostream>
int *pfmin(int *p, int n);
void main()
{
    int masB[5] = { 4, 8, 2, 6, 4 };
    (*pfmin(masB, 5))++;
    for (int i = 0; i < 5; i++)
        std::cout<<masB[i]<<' ';
}
int *pfmin(int *p, int n)
{
    int *pmin;
    for (pmin = p; n > 0; p++, n--)
        if (*p < *pmin)
            pmin = p;
    return pmin;
}
```

Пример. Пусть имеется массив **В**. С использованием функции пользователя определить минимальный элемент массива в подпрограмме и в основной программе увеличить его значение на 1.

3. Изучить использование *ссылки* как *результата* работы функции, выполнив программу, записанную в правой части.

Пример. Массив **А** содержит набор значений. Необходимо в подпрограмме определить минимальный элемент и изменить его на другое значение в основной программе.

```
#include <iostream>
using namespace std;

double &dmin(double A[], int size);
```

Опробовать работу программы с разными значениями массива A.

```
void main()
{
    double s;
    const int size = 5;
    double A[] = { 5, 4.1, 3, 0.2, 11 };
    s = dmin(A, size);
    cout << s<< endl;
    for (int i = 0; i < size; i++)
        cout << " " << A[i];
    cout << endl;
    dmin(A, size) = 1.0;    // изменение минимума на значение 1.0
    for (int i = 0; i < size; i++)
        cout << " " << A[i];
}

double &dmin(double A[], int size)
{
    int i, j = 0;
    for (i = 1; i < size; i++)
        if (A[j] > A[i])
            j = i;
    return A[j];
}
```

Ссылка — это псевдоним, и значит объект, на который она ссылается, существует после выполнения функции пользователя.

Если результат функции пользователя представлен в виде ссылки, то можно использовать вызов функции в левой части оператора присваивания.

4. Ознакомиться с использованием функций с результатом *логического* типа, опробовав работу программы в правой части.

```
#include <iostream>
using namespace std;
bool is_elem(int *pA, int n, int iV);

void main()
{   setlocale(LC_CTYPE, "Rus");
    bool t; int k;
    int A[] = { 5, 4, 3, 2, 11 };
    cout << "Введите число ";
    cin >> k;
    t = is_elem(A, sizeof(A)/sizeof(int), k);
    if (t == true)
        cout << "Число есть в массиве ";
    else
        cout << "Числа нет в массиве ";

}

bool is_elem(int *pA, int n, int iV)
{   bool bf = false;
    for (int i = 0; i < n; i++)
        if (pA[i] == iV)
        {   bf = true;
            break;
        }
    return bf;
}
```

Пример. Имеется массив целых чисел **A**. Определить, содержит ли он число, которое вводится с клавиатуры.

Результат, возвращаемый функцией **bool is_elem ()**, – это логическая переменная, которая может принимать два значения – **true** (если искомый элемент есть в массиве) и **false** (если элемент отсутствует).

5. В соответствии со своим вариантом написать главную функцию, в которой имеются вызовы *функций пользователя*, реализующих задачи из таблицы ниже. Ввод исходных данных и вывод результатов осуществить в главной функции, при этом использовать *динамические массивы*.

Для передачи параметров в функции пользователя и возвращения результатов применить *указатели* и *ссылки*.

№ варианта	Условие задачи
1	<p>1. Определить сумму элементов целочисленного массива, расположенных между первым максимальным и последним минимальным элементами.</p> <p>2. Даны две квадратные целочисленные матрицы. Если все числа положительны, то определить произведение этих матриц.</p>
2	<p>1. В массиве из целых чисел найти в процентах частоту появления каждого из k наиболее часто встретившихся чисел (k – натуральное число, не превосходящее числа элементов массива).</p> <p>2. Дана целочисленная прямоугольная матрица. Если есть в матрице нулевые элементы, то определить количество строк, содержащих их.</p>
3	<p>1. В массиве из целых чисел определить сумму элементов, расположенных между первым и последним нулевыми элементами</p> <p>2. Если есть в матрице строка, все элементы которой положительны, то найти сумму этих элементов. Уменьшить все элементы матрицы на эту сумму.</p>
4	<p>1. Создать одномерный массив, содержащий 15 элементов, наполнить его случайными значениями в интервале от 1 до 200. Определить сумму всех нечетных элементов массива.</p> <p>2. Проверить, есть ли в матрице хотя бы одна строка, содержащая положительный элемент и найти ее номер. Знаки элементов предыдущей строки изменить на противоположные.</p>

№ варианта	Условие задачи
5	<p>1. В одномерном массиве, содержащем 15 элементов, посчитать и вывести сумму всех четных элементов массива.</p> <p>2. Даны две целочисленные матрицы размером $n \times m$ и $m \times n$. Если в матрице отсутствуют нулевые элементы, то определить произведение этих матриц.</p>
6	<p>1. Найти сумму минимального и максимального элементов одномерного массива.</p> <p>2. Проверить, все ли строки матрицы содержат хотя бы один нулевой элемент. Если нет, то заменить значения всех отрицательных элементов матрицы на нулевые.</p>
7	<p>1. Создать одномерный массив, содержащий 15 элементов, заполнить его случайными значениями в интервале от 1 до 200. Определить произведение элементов массива с индексами от 2 до 7.</p> <p>2. Если есть в матрице столбец, все элементы которого отрицательны, то найти среднее арифметическое этих элементов. Вывести полученное значение из всех элементов матрицы.</p>
8	<p>1. В массиве вещественных чисел заменить отрицательные элементы их квадратами. Найти в полученном массиве индекс элемента, для которого сумма элементов, стоящих до него, наименее отличается от суммы элементов, стоящих после него.</p> <p>2. Проверить, есть ли в матрице хотя бы один столбец, содержащий отрицательный элемент, и найти его номер. Уменьшить элементы найденного столбца вдвое.</p>
9	<p>1. Создать одномерный массив, содержащий 15 элементов, заполнить его случайными значениями в интервале от 1 до 200. Определить произведение всех элементов массива, значения которых меньше 50.</p> <p>2. Дана целочисленная матрица A размерности $n \times m$ и вектор B размерности $m \times 1$. Если все числа отрицательны, то определить произведение этих массивов.</p>

№ варианта	Условие задачи
10	<p>1. Найти номера четных элементов, стоящих на нечетных местах в одномерном массиве.</p> <p>2. Дана целочисленная прямоугольная матрица. Если максимальное число встречается в матрице более одного раза, то вывести его значение и индексы, а также количество повторений.</p>
11	<p>1. Изменить одномерный массив, вычеркнув из него нечетные элементы.</p> <p>2. Дана целочисленная квадратная матрица. Если она не содержит отрицательных элементов, то определить сумму элементов в тех строках, где отрицательные элементы отсутствуют.</p>
12	<p>1. Найти количество отрицательных элементов, стоящих на четных местах в одномерном массиве.</p> <p>2. Проверить, есть ли в матрице хотя бы одна строка, содержащая отрицательный элемент, и найти ее номер. Все элементы столбца с таким же номером уменьшить вдвое.</p>
13	<p>1. Дан массив A вещественного типа, содержащий 20 положительных и отрицательных элементов. Сформировать массив B из положительных элементов массива A, имеющих четный индекс. Найти сумму квадратов элементов нового массива.</p> <p>2. Дана целочисленная квадратная матрица. Если минимальное значение находится на главной диагонали, то вывести его значение и индексы.</p>
14	<p>1. В одномерном массиве найти минимальный и максимальный элементы. Вычислить их разность.</p> <p>2. Если в матрице имеется столбец, все элементы которого положительны, то знаки элементов предыдущего столбца изменить на противоположные.</p>
15	<p>1. Дан одномерный массив. Определить, сколько отрицательных элементов находится после последнего минимального элемента.</p>

№ варианта	Условие задачи
	2. Проверить, все ли столбцы матрицы содержат хотя бы один положительный элемент. Если нет, то в первом столбце, не удовлетворяющем условию, заменить отрицательные элементы их модулями.
16	<p>1. В одномерном массиве найти максимальный из отрицательных элементов и поменять его местами с последним элементом массива.</p> <p>2. Проверить, все ли строки матрицы содержат хотя бы один отрицательный элемент. Если да, то изменить знаки всех элементов матрицы на обратные.</p>

[В начало практикума](#)

Стандарты оформления кода программы на языке C/C++

Стиль программирования – это набор правил и соглашений, используемых при написании исходного кода на некотором языке программирования. Наличие общего стиля программирования уменьшает вероятность появления ошибок на этапе набора текста, делает программу легко читаемой, что, в свою очередь, облегчает процессы отладки и внесения изменений.

Указанные ниже требования не являются единственно верными для написания кода, являются производными от широко распространённого стандарта, с которым можно ознакомиться в оригинале на следующих сайтах:

- http://www.opennet.ru/docs/RUS/coding_standard/ – стандарт кодирования GNU;
- <https://google-styleguide.googlecode.com/svn/trunk/cppguide.html> – Google C++ Style Guide;
- <http://geosoft.no/development/cppstyle.html> – C++ Programming Style Guidelines;
- <https://ru.wikipedia.org/wiki>, <http://habrahabr.ru/post/194752/> и др. – Стандарты оформления кода.

Отступы и фигурные скобки. Для всего проекта должен применяться единый стиль отступов. Отступы ставятся везде, где есть фигурные скобки. То есть в теле функций, циклах (do, while, for), операторах if и switch.

В настоящее время используются различные стили расстановки фигурных скобок.

1. Стил K&R или 1TBS (One True Bracing Style) Назван в честь Brian W. Kernighan и Dennis M. Ritchie:

```
if (condition) {
    content;
}
```

2. Стил Алмена (используется по умолчанию в Microsoft Visual Studio):

```
if (condition)
{
    content;
}
```

3. Стилль GNU (отступы всегда делаются по 4 символа на уровень, скобки находятся на половине отступа):

```
if (condition)
{
    content;
}
```

Рекомендуется каждый блок («if», «else», «while», «for», «do», «switch») заключать в скобки, даже если он содержит только одну строку. Сокращённая запись часто приводит к ошибкам, которые тяжело найти, т.к. если такой блок кода потребуется расширить, то наличие скобок становится обязательным, о чём не всегда вспоминает программист, дописавший код.

В булевских выражениях («if», «for», «while», «do» и первом операнде тернарного оператора "?") надо записывать символы равенства и неравенства в явном виде.

Длина строки. Длина строки не должна превышать 80 символов. Длинные выражения разделяются на части, при этом последующие части должны быть короче первой, и сдвинуты вправо (чтобы схожие логические объекты находились на одной вертикально прямой). Те же правила применяются к функциям с длинным списком аргументов и к текстовым строкам.

```
void fun(int a, int b, int c)
{
    if (condition)
    {
        printf("Warning: this is a long"
               "printf with 3 paramenters a: %u b: %u"
               "c: %u \n", a, b, c);
    }
    else
    {
        next_statement;
    }
}
```

Комментарии. Комментарии должны быть краткими и понятными, они нужны для облегчения чтения кода. При неправильном использовании комментариев можно только усложнить читабельность кода, например, комментируя те моменты, которые и так всем понятны.

Имена переменных, типов и функций. Существует несколько стилей названия переменных. Например, переменная **var_bell** может быть написана разными способами:

var_bell – стиль C: нижний регистр, знак подчеркивания;

VarBell – стиль Pascal: каждая подстрока в названии начинается с большой буквы;

varBell – стиль Java: первая строка начинается с маленькой буквы, все последующие – с большой.

Не имеет значения, какой стиль будет выбран – главное придерживаться в коде программы одного стиля.

Константы традиционно записываются в верхнем регистре (например, **YANDEX_BOT**).

При выборе имени переменной не так важна длина имени, как понятность. Длинные имена могут назначаться глобальным переменным, которые редко используются, но индексы массивов, появляющиеся в каждой строке цикла, не должны быть значительно сложнее, чем **i**. Использование таких имен, как **index** или **elementnumber**, не только усложняют набор, но и могут сделать менее понятными детали вычислений.

Рекомендуется называть переменные, функции и файлы со строчной буквы, а свои типы (в т.ч. классы) – с заглавной.

```
struct ListNode
{
    int value;
    ListNode *next;
};

int getNextValue();
void printList();
```

Функции-предикаты (булевы функции) следует именовать следующим образом: [глагол][предикат], где "глагол" – глагол бытия (to be) или обладания (to have) в соответствующей форме, а "предикат" – проверяемое свойство. Например,

```
bool isPrime(int n);
bool hasColor();
```

При вызове функций-предикатов запрещается сравнивать результат с логическими константами (**if (isPrime(n) == true)** – плохой код).

Все методы (функции), которые не изменяют состояние объекта, должны быть определены константными, например,

```
string getName() const
{
    return name;
}
```

Вывод данных. Каждый вывод данных на экран должен сопровождаться сообщением с пояснением того, что именно выводится.

Сокращения в коде программы. При написании кода стоит пользоваться упрощёнными конструкциями.

Неправильно	Правильно
b[j] = <выражение>; j++;	b[j++] = <выражение>;
a = <выражение>; return a; (лишняя переменная)	return <выражение>;
if (<условие>) {	return <условие>;

Неправильно	Правильно
<pre> return true; } else { return false; } (лишняя else-ветка) </pre>	
<pre> if (<условие>) { return true; } return false; (лишнее использование условного оператора) </pre>	<pre> return <условие>; </pre>
<pre> if (<условие>) { a = <выражение1>; } else { a = <выражение2>; } return a; (формирование лишней переменной внутри условного оператора) </pre>	<pre> if (<условие>) { return <выражение1>; } return <выражение2>; </pre>