

Методические рекомендации

Данный сборник предназначен для организации эффективного учебного процесса при изучении основ программирования на языке C++. Приведенные материалы прошли апробацию в учебной практике и показали свою результативность.

Базовые концепции C++

Задачи для урока

1. «Фруктовая арифметика»

В кафе продаются яблоки по 5 рублей и груши по 7 рублей. Посетитель хочет купить X яблок и Y груш. Напишите программу, которая запрашивает X и Y и выводит общую сумму к оплате.

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

int main() {
    int x, y;
    cin >> x >> y;
    cout << x * 5 + y * 7;
    return 0;
}
```

2. «Температурное путешествие»

Турист прилетел из России в Великобританию. Его интересует, какая будет температура по Фаренгейту, если в Москве сейчас X градусов по Цельсию. Напишите программу, которая выводит это значение по формуле $F = C * 1.8 + 32$.

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

int main() {
    double c;
    cin >> c;
    cout << c * 1.8 + 32;
    return 0;
}
```

3. «Волшебное число»

В древней игре нужно взять трехзначное число, сложить его цифры и получить "ключ к порталу". Напишите программу, которая запрашивает у пользователя трёхзначное число и выводит сумму его цифр.

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

int main() {
    int n;
    cin >> n;
    cout << n / 100 + n / 10 % 10 + n % 10;
    return 0;
}
```

4. «Магические функции»

Учёные из космической лаборатории вводят число — координату астероида. Нужно рассчитать: квадратный корень, экспоненту. Используйте библиотеку cmath. Вывод должен быть с пояснениями.

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

int main() {
    double num;
    cin >> num;
    cout << "Корень: " << sqrt(num) << endl;
    cout << "Экспонента: " << exp(num);
    return 0;
}
```

5. «Круглый агент»

Робот-курьер вычисляет, как округлить расстояние до ближайшего склада. Запросите вещественное число — расстояние — и выведите его округление вверх и вниз (ceil, floor).

```

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

int main() {
    double num;
    cin >> num;
    cout << "Вверх: " << ceil(num) << endl;
    cout << "Вниз: " << floor(num);
    return 0;
}

```

6. «Калькулятор дружбы»

Два ученика спорят, кто из них получит больше баллов. Введите два числа и выведите: сумму, разность, произведение, частное, остаток от деления.

Каждая операция — отдельная строка вывода.

```

#include <iostream>
using namespace std;

int main() {
    int a, b;
    cin >> a >> b;
    cout << a + b << endl;
    cout << a - b << endl;
    cout << a * b << endl;
    cout << a / b << endl;
    cout << a % b;
    return 0;
}

```

7. «Кодовая последовательность»

В архиве найден код из 6 чисел: 13 21 34 55 89 144. Напишите программу, которая выводит эти числа через пробел с помощью printf.

```

#include <cstdio>
using namespace std;

int main() {
    printf("13 21 34 55 89 144");
    return 0;
}

```

Домашнее задание

1. «Среднее по классу»

Учитель ввел три оценки ученика. Найдите их среднее арифметическое.

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

int main() {
    double a, b, c;
    cin >> a >> b >> c;
    cout << (a + b + c) / 3;
    return 0;
}
```

2. «Секундомер»

Вы записали, сколько секунд длился марафон. Программа должна перевести это в часы и минуты.

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

int main() {
    int seconds;
    cin >> seconds;
    cout << seconds / 3600 << " часов " << (seconds % 3600) / 60 << " минут";
    return 0;
}
```

3. «Курс валют»

Банкомат принимает рубли и показывает, сколько это долларов. Запросите сумму и выведите ее в долларах по курсу 1 USD = 87.3 RUB.

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

int main() {
    double rub;
    cin >> rub;
    cout << rub / 87.3 << " USD";
    return 0;
}
```

4. «Площадь круга»

Введите радиус окружности и найдите её площадь. Используйте $\pi = 3.1415$.

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

int main() {
    double r;
    cin >> r;
    cout << 3.1415 * r * r;
    return 0;
}
```

5. «Зеркальное число»

Пираты оставили код из трёх цифр. Нужно перевернуть его: 123 → 321. Напишите программу, которая запрашивает трёхзначное число и выводит его в обратном порядке

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

int main() {
    int n;
    cin >> n;
    cout << n % 10 << n / 10 % 10 << n / 100;
    return 0;
}
```

Ветвления

Задачи для урока

1. «Чётное или нечётное»

На работе установлена защита: он включается только, если введённое число чётное. Напишите программу, которая определяет, чётное ли целое число, и выводит «Да» или «Нет».

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

int main() {
    int num;
    cin >> num;
    if (num % 2 == 0) {
        cout << "Да";
    } else {
        cout << "Нет";
    }
    return 0;
}
```

2. «Турнир двух рыцарей»

На арене сражаются два рыцаря. У каждого — сила удара (введите два числа). Выведите имя победителя: «Первый», «Второй» или «Ничья».

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

int main() {
    int a, b;
    cin >> a >> b;
    if (a > b) cout << "Первый";
    else if (b > a) cout << "Второй";
    else cout << "Ничья";
    return 0;
}
```

3. «Метеостанция»

Введите температуру воздуха. Программа должна вывести:

- «Тепло», если температура > 20
- «Прохладно», если $0 \leq \text{температура} \leq 20$
- «Мороз», если температура < 0

```

#include <iostream>
using namespace std;

int main() {
    int temp;
    cin >> temp;
    if (temp > 20) cout << "Тепло";
    else if (temp >= 0) cout << "Прохладно";
    else cout << "Мороз";
    return 0;
}

```

4. «Контрольный пропуск»

В подземелье можно пройти только с кодом от 10 до 50. Запросите код и выведите: «Доступ разрешен» или «Доступ запрещен».

```

#include <iostream>
using namespace std;

int main() {
    int code;
    cin >> code;
    if (code >= 10 && code <= 50) {
        cout << "Доступ разрешен";
    } else {
        cout << "Доступ запрещен";
    }
    return 0;
}

```

5. «Мини-калькулятор»

Программа должна запрашивать два целых числа и символ операции (+, -, *, /). В зависимости от символа выводите результат соответствующего действия.

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

int main() {
    int a, b;
    char op;
    cin >> a >> op >> b;
    if (op == '+') cout << a + b;
    else if (op == '-') cout << a - b;
    else if (op == '*') cout << a * b;
    else if (op == '/') cout << a / b;
    return 0;
}
```

6. «Числовой знак»

Введите целое число. Выведите:

- «Положительное»
- «Отрицательное»
- «Равно нулю»

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

int main() {
    int num;
    cin >> num;
    if (num > 0) cout << "Положительное";
    else if (num < 0) cout << "Отрицательное";
    else cout << "Равно нулю";
    return 0;
}
```

7. «Зона безопасности»

Пользователь вводит координату X (в диапазоне от -100 до 100).
Определите, попадает ли она в "безопасную зону" от -30 до 30 включительно.


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

int main() {
    int x;
    cin >> x;
    if (x >= -30 && x <= 30) {
        cout << "Да";
    } else {
        cout << "Нет";
    }
    return 0;
}
```

Домашнее задание

1. «Год високосный?»

Введите год. Программа должна вывести, является ли он високосным.
Условия:

- Год делится на 4 и не делится на 100
- ИЛИ делится на 400

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

int main() {
    int year;
    cin >> year;
    if ((year % 4 == 0 && year % 100 != 0) || year % 400 == 0) {
        cout << "Да";
    } else {
        cout << "Нет";
    }
    return 0;
}
```

2. «Треугольник или нет?»

Введите три стороны. Выведите «Да», если можно построить треугольник (любые две стороны в сумме больше третьей).

```

#include <iostream>
using namespace std;

int main() {
    int a, b, c;
    cin >> a >> b >> c;
    if (a + b > c && a + c > b && b + c > a) {
        cout << "Да";
    } else {
        cout << "Нет";
    }
    return 0;
}

```

3. «Большое число»

Введите три числа. Программа должна вывести наибольшее из них.

```

#include <iostream>
using namespace std;

int main() {
    int a, b, c;
    cin >> a >> b >> c;
    if (a >= b && a >= c) {
        cout << a;
    } else if (b >= a && b >= c) {
        cout << b;
    } else {
        cout << c;
    }
    return 0;
}

```

4. «Цвет шахматной клетки»

В шахматной доске вводятся координаты клетки (от 1 до 8 по X и Y). Определите цвет клетки (белая или чёрная). Клетка (1,1) — чёрная.

```

#include <iostream>
using namespace std;

int main() {
    int x, y;
    cin >> x >> y;
    if ((x + y) % 2 == 0) {
        cout << "Чёрная";
    } else {
        cout << "Белая";
    }
    return 0;
}

```

5. «Угадай число»

Компьютер "загадывает" число 42 (жёстко задано в коде). Пользователь вводит свое число. Программа должна вывести:

- «Больше», если число больше 42
- «Меньше», если меньше
- «Угадал!», если равно

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

int main() {
    int num;
    cin >> num;
    if (num > 42) {
        cout << "Больше";
    } else if (num < 42) {
        cout << "Меньше";
    } else {
        cout << "Угадал!";
    }
    return 0;
}
```

Циклы

Задачи для урока

1. «Лестница чисел»

Робот строит лестницу из чисел. Он начинает с 1 и прибавляет по одному, пока не дойдёт до числа N. Введите число N и выведите все числа от 1 до N через пробел. Используйте while.

```

#include <iostream>
using namespace std;

int main() {
    int n;
    cin >> n;
    int i = 1;
    while (i <= n) {
        cout << i << " ";
        i++;
    }
    return 0;
}

```

2. «Пчёлы и мёд»

На каждой итерации пчела приносит 2 литра мёда. Сколько нужно итераций, чтобы собрать минимум X литров? Введите X и посчитайте количество шагов с помощью while.

```

#include <iostream>
using namespace std;

int main() {
    int x;
    cin >> x;
    int liters = 0;
    int steps = 0;
    while (liters < x) {
        liters += 2;
        steps++;
    }
    cout << steps;
    return 0;
}

```

3. «Чётные числа»

Введите число N. Выведите все чётные числа от 1 до N.

```

#include <iostream>
using namespace std;

int main() {
    int n;
    cin >> n;
    for (int i = 2; i <= n; i += 2) {
        cout << i << " ";
    }
    return 0;
}

```

4. «Магический квадрат»

Введите число N. Постройте квадрат NxN из чисел от 1 до N×N по строкам.

Например:

1 2 3

4 5 6

7 8 9

Используйте два for.

```

#include <iostream>
using namespace std;

int main() {
    int n;
    cin >> n;
    int num = 1;
    for (int i = 0; i < n; i++) {
        for (int j = 0; j < n; j++) {
            cout << num++ << " ";
        }
        cout << endl;
    }
    return 0;
}

```

5. «Треугольный числовой ряд»

Выведите на экран числа от 1 до N так:

1

2 3

4 5 6

...

Каждая строка содержит на 1 число больше. Используйте вложенные for.

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

int main() {
    int n;
    cin >> n;
    int num = 1;
    for (int i = 1; num <= n; i++) {
        for (int j = 1; j <= i && num <= n; j++) {
            cout << num++ << " ";
        }
        cout << endl;
    }
    return 0;
}
```

Домашнее задание

1. «Факториал»

Введите число N. Найдите его факториал N! с помощью цикла while.

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

int main() {
    int n;
    cin >> n;
    int result = 1;
    int i = 1;
    while (i <= n) {
        result *= i;
        i++;
    }
    cout << result;
    return 0;
}
```

2. «Мощь степени»

Введите два числа A и B. Выведите A в степени B (без использования pow)

с помощью цикла while.

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

int main() {
    int a, b;
    cin >> a >> b;
    int result = 1;
    int i = 0;
    while (i < b) {
        result *= a;
        i++;
    }
    cout << result;
    return 0;
}
```

3. «Обратный порядок»

Введите число N. Выведите все числа от N до 1 в обратном порядке.

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

int main() {
    int n;
    cin >> n;
    while (n >= 1) {
        cout << n << " ";
        n--;
    }
    return 0;
}
```

4. «Фигурная сетка»

Постройте сетку размера N×M из символов @ (вводятся N и M).

Пример (N = 3, M = 5):

@@@@@

@@@@@

@@@@@

```

#include <iostream>
using namespace std;

int main() {
    int n, m;
    cin >> n >> m;
    for (int i = 0; i < n; i++) {
        for (int j = 0; j < m; j++) {
            cout << "@ ";
        }
        cout << endl;
    }
    return 0;
}

```

5. «Обратная лесенка»

Введите число N. Постройте лесенку из символов #, начиная с N и уменьшая до 1:

```
#####
```

```
####
```

```
###
```

```
##
```

```
#
```

```

#include <iostream>
using namespace std;

int main() {
    int n;
    cin >> n;
    while (n >= 1) {
        for (int i = 0; i < n; i++) {
            cout << "#";
        }
        cout << endl;
        n--;
    }
    return 0;
}

```


Массивы

Задачи для урока

1. «Случайный генератор»

Учёный генерирует 10 случайных чисел от 1 до 100. Выведите их в строку через пробел. Используйте `rand() % 100 + 1` и массив.
(сделать ввод размера *N*)

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

int main() {
    srand(time(0));
    int arr[10];
    for (int i = 0; i < 10; i++) {
        arr[i] = rand() % 100 + 1;
        cout << arr[i] << " ";
    }
    return 0;
}
```

2. «Сумма положительных»

Введите массив из *N* целых чисел. Выведите сумму всех положительных чисел в массиве.

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

int main() {
    int n;
    cin >> n;
    int arr[n], sum = 0;
    for (int i = 0; i < n; i++) {
        cin >> arr[i];
        if (arr[i] > 0) sum += arr[i];
    }
    cout << sum;
    return 0;
}
```

3. «Наибольшее и наименьшее»

Введите массив из *N* чисел. Найдите максимальное и минимальное число.

```

#include <iostream>
using namespace std;

int main() {
    int n;
    cin >> n;
    int arr[n];
    for (int i = 0; i < n; i++) cin >> arr[i];

    int min = arr[0], max = arr[0];
    for (int i = 1; i < n; i++) {
        if (arr[i] < min) min = arr[i];
        if (arr[i] > max) max = arr[i];
    }
    cout << "Min: " << min << "\nMax: " << max;
    return 0;
}

```

4. «Четные — налево!»

Введите массив из N целых чисел. Выведите сначала все четные числа, затем нечетные, сохранив порядок.

Пример:

Ввод: 1 4 2 3 5 6 → Вывод: 4 2 6 1 3 5

```

#include <iostream>
using namespace std;

int main() {
    int n;
    cin >> n;
    int arr[n];
    for (int i = 0; i < n; i++) cin >> arr[i];

    // Сначала выводим четные
    for (int i = 0; i < n; i++)
        if (arr[i] % 2 == 0) cout << arr[i] << " ";

    // Затем нечетные
    for (int i = 0; i < n; i++)
        if (arr[i] % 2 != 0) cout << arr[i] << " ";

    return 0;
}

```

5. «Уникальные значения»

Введите массив из N чисел. Удалите все повторы, оставив только первые вхождения.

(Не использовать встроенные структуры, только массивы и индексы)

```

#include <iostream>
using namespace std;

int main() {
    int n;
    cin >> n;
    int arr[n];
    for (int i = 0; i < n; i++) cin >> arr[i];

    cout << arr[0] << " ";
    for (int i = 1; i < n; i++) {
        bool isUnique = true;
        for (int j = 0; j < i; j++) {
            if (arr[i] == arr[j]) {
                isUnique = false;
                break;
            }
        }
        if (isUnique) cout << arr[i] << " ";
    }
    return 0;
}

```

6. «Второе по величине»

Найдите второе по величине число в массиве из N чисел.

Пример:

Ввод: 5 1 8 3 6 → Вывод: 6

```

#include <iostream>
using namespace std;

int main() {
    int n;
    cin >> n;
    int arr[n];
    for (int i = 0; i < n; i++) cin >> arr[i];

    int max1 = -1e9, max2 = -1e9;
    for (int i = 0; i < n; i++) {
        if (arr[i] > max1) {
            max2 = max1;
            max1 = arr[i];
        }
        else if (arr[i] > max2 && arr[i] != max1) {
            max2 = arr[i];
        }
    }
    cout << max2;
    return 0;
}

```

Домашнее задание

1. «Сумма всех»

Введите массив из N чисел. Найдите сумму всех элементов массива.

```

#include <iostream>
using namespace std;

int main() {
    int n;
    cin >> n;
    int arr[n], sum = 0;
    for (int i = 0; i < n; i++) {
        cin >> arr[i];
        sum += arr[i];
    }
    cout << sum;
    return 0;
}

```

2. «Количество нулей»

Сколько раз в массиве из N чисел встречается 0?

```

#include <iostream>
using namespace std;

int main() {
    int n;
    cin >> n;
    int arr[n], count = 0;
    for (int i = 0; i < n; i++) {
        cin >> arr[i];
        if (arr[i] == 0) count++;
    }
    cout << count;
    return 0;
}

```

3. «Обратный порядок»

Введите массив из N чисел. Выведите его в обратном порядке.

```

#include <iostream>
using namespace std;

int main() {
    int n;
    cin >> n;
    int arr[n];
    for (int i = 0; i < n; i++) cin >> arr[i];

    for (int i = n-1; i >= 0; i--) cout << arr[i] << " ";
    return 0;
}

```

4. «Среднее арифметическое»

Найдите среднее значение чисел в массиве. Выведите его с двумя знаками после запятой.

```

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

int main() {
    int n;
    cin >> n;
    int arr[n];
    double sum = 0;
    for (int i = 0; i < n; i++) {
        cin >> arr[i];
        sum += arr[i];
    }
    cout << fixed << setprecision(2) << sum / n;
    return 0;
}

```

5. «Разворот»

Разверните массив “вперёд-назад” — первый с последним, второй с предпоследним и т. д. Используйте только один дополнительный временный элемент.

```

#include <iostream>
using namespace std;

int main() {
    int n;
    cin >> n;
    int arr[n];
    for (int i = 0; i < n; i++) cin >> arr[i];

    for (int i = 0; i < n/2; i++) {
        int temp = arr[i];
        arr[i] = arr[n-1-i];
        arr[n-1-i] = temp;
    }

    for (int i = 0; i < n; i++) cout << arr[i] << " ";
    return 0;
}

```

Умные строки и массивы символов

Задачи для урока

1. «Гласные и согласные»

Школьнику дали задание посчитать гласные и согласные в тексте. Введите строку (латиницей, строчные буквы) и выведите, сколько в ней:

- гласных: а, е, і, о, u
- согласных: остальные буквы

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

int main() {
    string s;
    getline(cin, s);
    int vowels = 0, consonants = 0;
    for (int i = 0; i < s.length(); i++) {
        char c = tolower(s[i]);
        if (isalpha(c)) {
            if (c == 'a' || c == 'e' || c == 'i' || c == 'o' || c == 'u')
                vowels++;
            else {
                consonants++;
            }
        }
    }
    cout << "Гласных: " << vowels << "\nСогласных: " << consonants;
    return 0;
}
```

2. «Обратный порядок»

Введите строку и выведите ее в обратном порядке.

Пример: hello → olleh

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

int main() {
    string s;
    getline(cin, s);
    for (int i = s.length() - 1; i >= 0; i--) {
        cout << s[i];
    }
    return 0;
}
```

3*. «IP-детектив»

Введите строку — возможный IP-адрес. Программа должна проверить:

- состоит ли он из четырёх чисел, разделённых точками
 - находятся ли числа в диапазоне от 0 до 255
- Выведите Корректный IP или Некорректный IP

4. «Удалить повторы»

Введите строку. Удалите все повторяющиеся символы, оставив только первые вхождения.

Пример: banana → ban

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

int main() {
    string s;
    getline(cin, s);
    string result;
    for (int i = 0; i < s.length(); i++) {
        bool found = false;
        int j = 0;
        while (j < result.length() && !found) {
            if (s[i] == result[j]) {
                found = true;
            }
            j++;
        }
        if (!found) {
            result += s[i];
        }
    }
    cout << result;
    return 0;
}
```

5. «Цифры в строке»

Введите строку. Найдите и выведите все цифры, которые в ней встречаются.

Пример: abc123def4 → 1 2 3 4


```

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

int main() {
    string s;
    getline(cin, s);
    for (char c : s) {
        if (isdigit(c)) {
            cout << c << " ";
        }
    }
    return 0;
}

```

6. «Строка-зеркало»

Введите строку. Программа должна создать новую строку, в которой:

- все буквы на чётных позициях — заглавные
 - все буквы на нечётных позициях — строчные
- Пример: hello → HeLlO

```

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

int main() {
    string s;
    getline(cin, s);
    for (int i = 0; i < s.size(); i++) {
        if (i % 2 == 0) {
            s[i] = toupper(s[i]);
        } else {
            s[i] = tolower(s[i]);
        }
    }
    cout << s;
    return 0;
}

```

Домашнее задание

1. «Подстрочная магия»

Введите строку и подстроку. Определите, сколько раз подстрока встречается в строке.

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

int main() {
    string s, sub;
    getline(cin, s);
    getline(cin, sub);
    int count = 0;
    size_t pos = 0;
    while ((pos = s.find(sub, pos)) != string::npos) {
        count++;
        pos += sub.length();
    }
    cout << count;
    return 0;
}
```

2. «Только буквы!»

Введите строку. Удалите из нее все символы, кроме букв латинского алфавита.

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

int main() {
    string s;
    getline(cin, s);
    string result;
    for (char c : s) {
        if (isalpha(c)) {
            result += c;
        }
    }
    cout << result;
    return 0;
}
```

3. «Через тире»

Введите слово. Выведите его так, чтобы между буквами было тире.
Пример: hello → h-e-l-l-o

```

#include <iostream>
using namespace std;

int main() {
    string s;
    getline(cin, s);
    for (int i = 0; i < s.length(); i++) {
        cout << s[i];
        if (i != s.length() - 1) {
            cout << "-";
        }
    }
    return 0;
}

```

4. «Слова наоборот»

Введите предложение. Выведите все слова в нём в обратном порядке, но сами слова не разворачивать.

Пример: I love C++ → C++ love I

```

#include <iostream>
using namespace std;

int main() {
    string s;
    getline(cin, s);
    string word;
    string result;
    for (int i = s.length() - 1; i >= 0; i--) {
        if (s[i] == ' ') {
            result = word + " " + result;
            word = "";
        } else {
            word = s[i] + word;
        }
    }
    result = word + " " + result;
    cout << result;
    return 0;
}

```

5. «Цифры словами»

Введите строку. Замените все цифры на их текстовые эквиваленты (1 → один, 2 → два и т. д.).

Пример: abc2x1 → abcdваждыодин

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

int main() {
    string digitWords[] = {"ноль", "один", "два", "три", "четыре",
                           "пять", "шесть", "семь", "восемь", "девять"};
    string s;
    getline(cin, s);
    string result;
    for (char c : s) {
        if (isdigit(c)) {
            result += digitWords[c - '0'];
        } else {
            result += c;
        }
    }
    cout << result;
    return 0;
}
```