

Занятие 1. Знакомство с C++

1. Вывод данных на экран.

Ниже приведен пример программы «Hello, World». Измените программу таким образом, чтобы она выводила ваши фамилию, имя, возраст, хобби. Для подключения русского языка используйте команду: `setlocale(0, ""); // русский язык`

```
#include <iostream>
using namespace std;
int main()
{
    cout << "Hello, world!"<<endl;
    return 0;
}
```

2. Объявление и инициализация переменных. Арифметические операции. Приведен пример программы с объявлением и инициализацией переменных.

```
#include <iostream>
using namespace std;
int main()
{
    int a=101;
    int b = 10;
    int c = a + b;
    cout <<" a + b = "<<c<< endl;
    return 0;
}
```

1

Используя арифметические операции

- + сложение;
- - вычитание;
- * умножение;
- / деление;
- % остаток от деления

найдите значения выражений $a - b$, $a * b$, a / b , $a \% b$ и выведите результат на печать. Прodelайте это же с переменными типа `double` (кроме операции `%`).

3. Математические функции C++. Библиотека `cmath`.

Подключите библиотеку `#include <cmath>`

```
#include <iostream>
#include <cmath>
using namespace std;
int main()
{
    double x = 2.0;
    double y = 3.0;

    return 0;
}
```

Вычислите и выведите на печать:

x^y , $\sin(x)$, $\sin(x)^2 + \cos(x)^2$, $\exp(x)$, $\arcsin(\sin(x))$, $\log(x)$, $\exp(x)$, $\log(\exp(x))$, $\exp(\log(x))$.

Функция	Описание
abs(a)	модуль или абсолютное значение от a
sqrt(a)	корень квадратный из a, причём a не отрицательно
pow(a, b)	возведение a в степень b
ceil(a)	округление a до наименьшего целого, но не меньше чем a
floor(a)	округление a до наибольшего целого, но не больше чем a
fmod(a, b)	вычисление остатка от a/b
exp(a)	вычисление экспоненты e ^a
sin(a)	a задаётся в радианах
cos(a)	a задаётся в радианах
log(a)	натуральный логарифм a(основанием является экспонента)
log10(a)	десятичный логарифм a
asin(a)	арксинус a, где -1.0 < a < 1.0

4. Логические выражения и операции сравнения.

Заданы целочисленные переменные. Выполните их сравнение и выведите на экран результаты.

```
#include <iostream>
using namespace std;
int main()
{
    setlocale(0, "");
    double a = 2.0;
    double b = 3.0;
    cout << a<<" равно "<<b<<" это " << (a == b) << endl;
    return 0;
}
```

2

Условие	Отношение
a == b	a равно b
a != b	a не равно b
a > b	a больше b
a < b	a меньше b
a >= b	a больше или равно b
a <= b	a меньше или равно b

5. Логические выражения и операции сравнения.

Обратите внимание на то, как записывается операция равенства – это ==. Также обратите внимание на то, что логические выражения записываются в скобках.

Исправьте ошибки в коде, приведенном ниже. Закомментированная строка должна выполняться.

```
#include <iostream>
using namespace std;
int main()
{
    double a = 2.0;
    double b = 3.0;
    cout << (a=b) << endl;
    cout << a << endl;
    //cout << a<b << endl;
    return 0;
}
```

6. Логические операции

Сформируйте и выведите на печать таблицы истинности для логических операций: И, ИЛИ, НЕ.

Операция	Пример применения
И &&	(5>3)&&(4<7)
или	(0==0) (3>7)
НЕ !	!(5==4)

7. Поразрядные логические операции

Операция	Обозначение
поразрядное И	&
поразрядное исключающее ИЛИ	^
поразрядное включающее ИЛИ	
~ – поразрядная инверсия (НЕ).	~

Бит в позиции 0 первого операнда сравнивается с битом в позиции 0 второго операнда. Затем бит в позиции 1 первого операнда сравнивается с битом в позиции 1 второго операнда. Так сравниваются все биты целочисленных операндов.

Выведите на экран все возможные комбинации поразрядных логических операций для заданных b1 и b2.

```
#include <iostream>
using namespace std;
int main()
{
    unsigned short int b1 = 41;
    unsigned short int b2 = 17;
    cout << (b1|b2) << endl;

    return 0;
}
```

3

8. Операции сдвига >> и <<

Операция и пример	Действие
<< - сдвиг влево (1<<5) // 2 в 5-й степени	сдвиг влево значения операнда на заданное количество бит. Операнд размещается слева от знака операции. Число сдвигаемых бит указывается справа от знака операции. Выдвижные биты переходят в старшие разряды, «входят» нулевые биты.
>> - сдвиг вправо (8>>2) // получим 2	сдвиг вправо значения операнда на заданное количество бит. Операнд размещается слева от знака операции. Количество сдвигаемых бит размещается справа от знака операции. Выдвижные биты теряются, а «входят» нулевые биты.

Если в программе необходимо, чтобы операция умножения целочисленных операндов на 2, 4, 8, происходила максимально быстро, то целесообразно использовать операцию сдвига влево <<.

Это касается и случаев, когда нужно быстро поделить целочисленный операнд на 2, 4, 8 и т.д. В этих случаях рекомендуется использовать сдвиг вправо >>.

```
#include <iostream>
using namespace std;
int main()
{
    int a = 1;
    int b = a << 1;
    cout << b << endl;
    int c = b << 2;
    cout << c << endl;
    return 0;
}
```

Получите значения 2^{10} , и $204800/256$ с помощью обычных вычислений и с помощью операций сдвига.

9. Операции инкремента и декремента постфиксные и префиксные

Операция	Обозначение
Префиксный инкремент	++a
Постфиксный инкремент	a++
Префиксный декремент	--a
Постфиксный декремент	a--

Выполните код, приведенный ниже, и проанализируйте отличие постфиксных инкремента и декремента от префиксных. Выведите сделанные выводы на экран.

```
#include <iostream>
using namespace std;
int main()
{
    int a = 100;
    cout << "a = " << a << endl;
    cout << "a++ " << a++ << endl;
    cout << "++a " << ++a << endl << endl;
    cout << "a = " << a << endl;
    cout << "a-- " << a-- << endl;
    cout << "--a " << --a << endl;
    return 0;
}
```

10. Команда ввода cin из библиотеки iostream

Измените код программы таким образом, чтобы на экран выводилась сумма трех чисел, введенных с клавиатуры.

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

int main()
{
    setlocale(0, "");
    int a;
    cout << "Введите целое число : ";
    cin >> a;
    double b;
    cout << "Введите вещественное число : ";
    cin >> b;
    return 0;
}
```

Задания для закрепления пройденного материала

1. Найдите площадь треугольника, если заданы три его стороны a , b , c . Используйте формулу Герона: $S = \sqrt{p(p-a)(p-b)(p-c)}$, p - полупериметр.
2. Создайте таблицу степеней двойки, используя операции поразрядного сдвига.
3. Заданы два целых числа, выведите 1, если первое число больше второго, выведите ноль, если 2-е число меньше или равно второму.
4. Заданы два целых числа. С помощью операции инкремента увеличьте первое число на 3, а второе уменьшите на 2.
5. Посчитайте \sin и \cos в градусах для следующих углов: 36, 74, 83. Выполните сначала проверку для косинуса 60 градусов. Для расчетов используйте значение π

```
const double PI = 3.1415926535897932384626433832795;
```

6. Задано целое трехзначное число. Посчитайте сумму цифр его составляющих.
7. Задано целое трехзначное число. Переставьте в нем первую и последнюю цифры.
8. Заданы три логические переменные

```
bool a=true;
bool b = true;
bool c = false;
```

Выведите на экран значения выражений:

$\text{NOT}((a \text{ OR } b) \text{ AND } (c))$, $\text{NOT}((a \text{ AND } b) \text{ OR } (c))$

Затем опишите новые переменные

```
int a1 = 5;
int b1 = 3;
int c1 = 0;
```

И выведите на экран значения аналогичных логических выражений.

9. Сравните действие операций деления для целых и вещественных переменных.
10. Инициализируйте две вещественные переменные. Их сумму присвойте целочисленной переменной и выведите результат на экран.
11. Инициализируйте две целочисленные переменные. Их сумму присвойте вещественной переменной и выведите результат на экран.
12. Создайте программу, содержащую следующий код:

```
unsigned long int a = - 5;
int b = 3;
cout << (a+b) << endl;
```

Объясните получившийся результат.

Контрольные вопросы

1. Чем отличается инициализация переменной от описания?
2. Как найти остаток от деления нацело?
3. Как возвести число в степень 23?
4. Что обозначает команда endl?
5. Как записывается операция «не равно»?
6. Как действует операция >> сдвига вправо?
7. Что такое инкремент и декремент?
8. Чем операция a++ отличается от операции ++a?
9. Что произойдет, если целочисленной переменной присвоить вещественное значение?
10. Какое число будет выведено на экран, после выполнения фрагмента кода:
cout << 7/2 <<endl;?
11. Для чего нужна строка #include <iostream>?
12. Какую функцию выполняет команда using namespace std;?
13. Есть ли ошибка в строке: cout << 4==5 <<endl;?
14. Что будет выведено после выполнения фрагмента кода: cout << (1<<1) <<endl; ?
15. Когда нужно ставить точку с запятой в коде C++?
16. Чем операция && отличается от &?
17. Какие типы данных C++ вы уже знаете?
18. Что будет выведено после выполнения фрагмента кода: cout << (0.367 && 0.123) <<endl;?
19. Исправьте ошибку в строке: cout << 3 <= 4 <<endl;?
20. Что будет выведено после выполнения фрагмента кода: cout << (!(5)!=6) <<endl;?
21. Какой командой можно подключить русский язык для вывода на экран?
22. Исправьте ошибку в строке: cout << (67 => 35) <<endl;?
23. Исправьте ошибку в строке: cout >>"Hello, World!">>endl;?
24. Исправьте ошибки во фрагменте программы:


```
int A = 2
int b = 3;
cout <<a+b endl;
```
25. Что будет выведено после выполнения фрагмента кода: cout <<(5<<3)<<endl;?