

Уровни языков программирования

Уровни языков программирования:

1. Машинные коды
2. Операции процессора
3. 3.1. Ассемблер – Windows
3.2. C/Shell – Unix
3.3. C++, Fortran – среднеуровневые языки
4. Java, Python...
5. Спец языки - JavaScript, HTML

Задача (+тех. средства) → Написание кода → Проверка синтаксиса →
Лексический анализ (поиск лексем) → Синтаксический анализ → Сборка
→ Программа/Бинарный код

#include <iostream> Директивы предпроцессора, подключение внешних файлов, библиотек

```
using namespace std; Пространство имён std
int main( )
{
    int x; - объявление переменной
    x=42; - присвоение значения
    cout << x;
    cout << «WTF!»; - вывод фразы
    return 0;
}
```

Операторы:

1. ; – пустой оператор
2. – составной оператор
3. +, -, *, /, %, ++, -- арифметические операции
4. ==, !=, >, <, >=, <= – сравнение
5. &&, ||, ! – логические операции
6. = – присвоение
7. int, bool, double... – обозначение типов
8. if (Логическое выражение) оператор 1
else оператор 2

9. 9.1. while (Логическое выражение) оператор 1
- 9.2. do оператор 1 while (Логическое выражение)
- 9.3. for (инициализация; логическое выражение; завершение действия)


```

x=0;
for (int i=0; i<100; i++)
x=x+i;

```
10. cin»a»b; – ввод
 cout«a«b; – вывод
11. break, continue – выход из цикла (желательно не использовать)

```

int main( )
{
int a=0, l=1;
for (int i=0; i<100; i++)
int temp; //temp – временная переменная существующая только внутри
цикла
cin»temp;
a=a+temp;
l=b*temp;
}
cout« a«b;
return;
}

```

```

int f(int a, int b)
{
if (b!=0)
return f(b,a%b)
return a;
} – функция нахождение НОД

```

```

int a(int n)
{
int s=1;
for (int i=1; i<=n; i++)
s=s*I;
return s;
} – факториал

```

```

int a(int n)
{

```

```
if (n==1) return 1;  
return n*a(n-1);  
} – рекурсивный факториал
```

```
s*=I быстрее s=s*i  
s++ быстрее s+=1 быстрее s=s+1;
```