Типы данных в языке С++

- целочисленный тип

-вещественный

-логический

-символьный

-void

**Целочисленный тип** ->

1. short

2. int

3. long

4. long long

sighed – Диапазон числа K = -2n-1 + ... + 2n-1 -1

unsigned – K = 0 ... +2n – 1

sizeof(int) – разрядность в байтах

int8\_t – целочисленное число под которое отводиться 8 бит

1 бит – signed, диапазон -128 – 127

uint8\_t – целочисленное число под которое отводиться 8 бит

unsigned диапазон 0 – 255

**Вещественный** ->

1. float 4 байта +- 1.18 \* 10-38 … +-3.4\*1038

2. double 8 байт +-2.23 \* 10-308 … +- 1.8 \* 10308

3. long double 8 байт/12 байт +-3.36 \* 10-4932 … +- 1.18 \* 104932

**Cимвольный тип**->

1. char – 0…255

ASCII ->

* 0…31 – спецсимволы
* 32…127 –> символьный для вывода
* 65…90 –> “A” … “Z”
* 97…122 –> “a” …”z”

**Инициализация**

char a = ‘1’; -> Копирование

char b(‘#’) ; прямая

char c{‘w’}; универсальная

char a = 97; (a = ‘a’)

cout <<int(a)<<static\_cast<int>(b)<<static\_cast<int>(c)<<endl; -> вывод номер в ASCII таблицы ‘1’ = 49

void fun(void) -> ничего не возвращает

**Переменные**

  int a, b;

  int a1 = 5, b1 = 10;

  char a2(97), a3('Z');

  a1 = (a + b) / a2;

**Константы**

const int MAX = 100;

int count MIN = 200;

int const IND – MAX – MIN;

**compili1 run time**

int n;

cin >> n;

const int INTMAX = n; инициализация константы во время работы

**Задание констант**

#define <идентификатор><текст>

#define <PI><3.14>

**Подток ввода выводы**

# include <iostream>

cin -> входной поток

cout -> выходной поток

cerr -> поток вывода сообщений об ошибки

<< - получить входной поток

>> - получить в выходной поток

Std(библиотека)::(операция области видимости)cout << n;

Чтобы не писать std, его можно задать в начале программы

using namespace std;

<iomanip> - библиотека для манипуляции с выводом вещественных чисел

Cout.width(15) – отводим 15 знаков для вывода

Fixed – установка точного количества знаков после запятой