## Цель:

1. Восстановление знаний о языке C/C++, а так же бреде разработки Visual Studio

Постановка задачи:

1. Необходимо написать программу, которая выведет все простейшие типы данных, их размеры в байтах, а так же диапазон значений (т.е. минимальное и максимальное значение для каждого типа данных).

Код программы

#include <stdio.h>

#include <stdbool.h>

#include <limits.h>

#include <float.h>

void printChar() {

printf("The minimum value of CHAR = %d\n", CHAR\_MIN);

printf("The maximum value of CHAR = %d\n", CHAR\_MAX);

printf("The size of CHAR = %d\n\n", sizeof(char));

printf("The minimum value of SIGNED CHAR = %d\n", SCHAR\_MIN);

printf("The maximum value of SIGNED CHAR = %d\n", SCHAR\_MAX);

printf("The size of SIGNED CHAR = %d\n\n", sizeof(signed char));

printf("The minimum value of UNSIGNED CHAR = %d\n", 0);

printf("The maximum value of UNSIGNED CHAR = %d\n", UCHAR\_MAX);

printf("The size of UNSIGNED CHAR = %d\n\n", sizeof(unsigned char));

}

void printShort() {

printf("The minimum value of SHORT = %d\n", SHRT\_MIN);

printf("The maximum value of SHORT = %d\n", SHRT\_MAX);

printf("The size of SHORT = %d\n\n", sizeof(short));

printf("The minimum value of UNSIGNED SHORT = %d\n", 0);

printf("The maximum value of UNSIGNED SHORT = %d\n", USHRT\_MAX);

printf("The size of UNSIGNED SHORT = %d\n\n", sizeof(unsigned short));

}

void printInt() {

printf("The minimum value of INT = %d\n", INT\_MIN);

printf("The maximum value of INT = %d\n", INT\_MAX);

printf("The size of INT = %d\n\n", sizeof(int));

printf("The minimum value of UNSIGNED INT = %u\n", 0);

printf("The maximum value of UNSIGNED INT = %u\n", UINT\_MAX);

printf("The size of UNSIGNED INT = %d\n\n", sizeof(unsigned int));

}

void printLong() {

printf("The minimum value of LONG = %li\n", LONG\_MIN);

printf("The maximum value of LONG = %li\n", LONG\_MAX);

printf("The size of LONG = %d\n\n", sizeof(long));

printf("The minimum value of UNSIGNED LONG = %lu\n", 0);

printf("The maximum value of UNSIGNED LONG = %lu\n", ULONG\_MAX);

printf("The size of UNSIGNED LONG = %d\n\n", sizeof(unsigned long));

printf("The minimum value of LONG LONG = %lli\n", LLONG\_MIN);

printf("The maximum value of LONG LONG = %lli\n", LLONG\_MAX);

printf("The size of LONG LONG = %d\n\n", sizeof(long long));

printf("The minimum value of UNSIGNED LONG LONG = %llu\n", 0);

printf("The maximum value of UNSIGNED LONG LONG = %llu\n", ULLONG\_MAX);

printf("The size of UNSIGNED LONG LONG = %d\n\n", sizeof(unsigned long long));

}

void printFloat() {

printf("The minimum value of FLOAT = %f\n", FLT\_MIN);

printf("The maximum value of FLOAT = %f\n", FLT\_MAX);

printf("The size of FLOAT = %d\n\n", sizeof(float));

}

void printDouble() {

printf("The minimum value of DOUBLE = %f\n", DBL\_MIN);

printf("The maximum value of DOUBLE = %f\n", DBL\_MAX);

printf("The size of DOUBLE = %d\n\n", sizeof(double));

printf("The minimum value of LONG DOUBLE = %Lf\n", LDBL\_MIN);

printf("The maximum value of LONG DOUBLE = %Lf\n", LDBL\_MAX);

printf("The size of LONG DOUBLE = %d\n\n", sizeof(long double));

}

void printBool() {

printf("The minimum value of BOOL(false) = %d\n", false);

printf("The maximum value of BOOL(true) = %d\n", true);

printf("The size of BOOL = %d\n\n", sizeof(bool));

}

int main(void) {

printChar();

printShort();

printInt();

printLong();

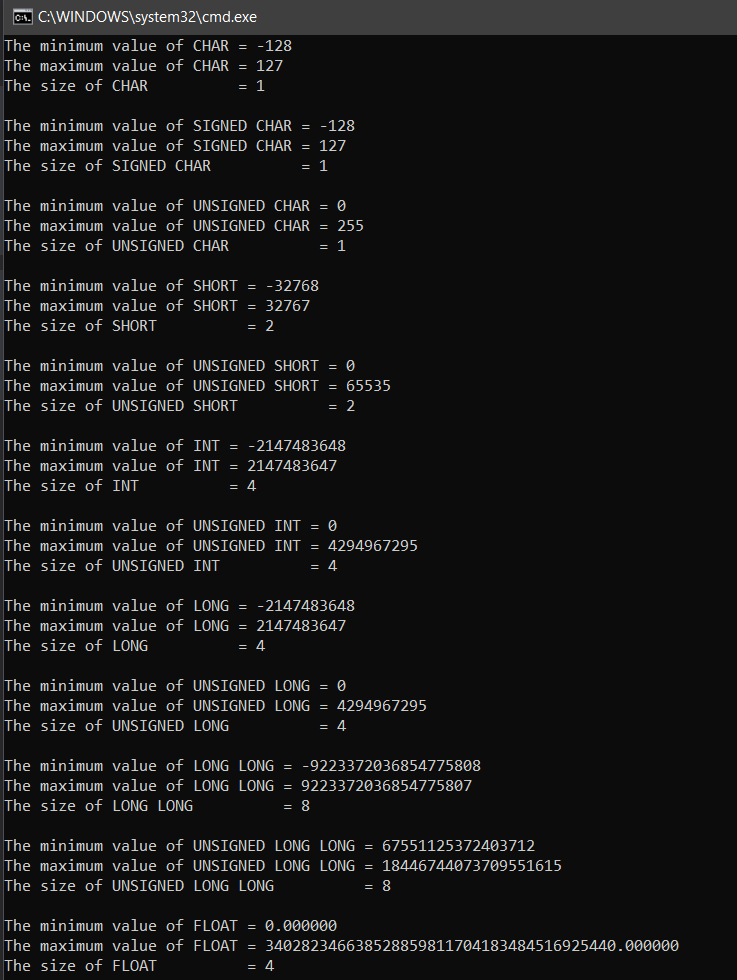
printFloat();

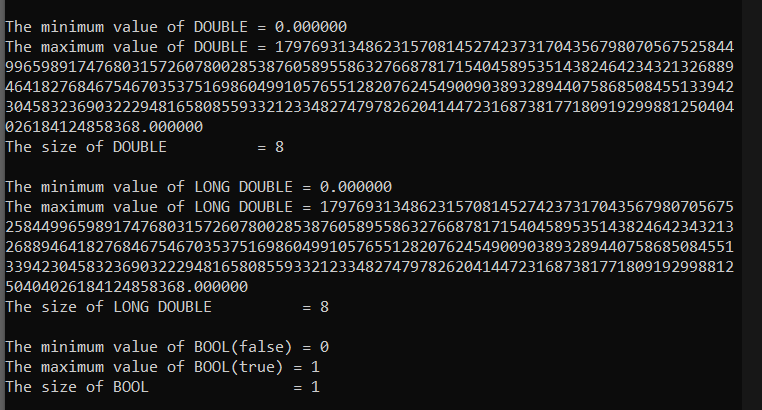
printDouble();

printBool();

}

Результат





Выводы:

Большинство из типов данных формируется с помощью одного из четырёх арифметических спецификаторов типа, (char, int, float и double), и опциональных спецификаторов (signed, unsigned, short и long). Хотя стандартом установлен диапазон, вычисляемый по формуле от −(2n−1−1) до 2n−1−1, все известные компиляторы допускают диапазон от −(2n−1) до 2n−1−1, где n — разрядность типа.

* bool: логический тип. Может принимать одну из двух значений true (истина) и false (ложь)
* char: представляет один символ в кодировке ASCII. Занимает в памяти 1 байт (8 бит). Может хранить любое значение из диапазона от -128 до 127, либо от 0 до 255
* signed char: представляет один символ. Занимает в памяти 1 байт (8 бит). Может хранить любой значение из диапазона от -128 до 127
* unsigned char: представляет один символ. Занимает в памяти 1 байт (8 бит). Может хранить любой значение из диапазона от 0 до 255
* short: представляет целое число в диапазоне от –32768 до 32767. Занимает в памяти 2 байта (16 бит).
* unsigned short: представляет целое число в диапазоне от 0 до 65535. Занимает в памяти 2 байта (16 бит).
* int: представляет целое число. В зависимости от архитектуры процессора может занимать 2 байта (16 бит) или 4 байта (32 бита). Диапазон предельных значений соответственно также может варьироваться от –32768 до 32767 (при 2 байтах) или от −2 147 483 648 до 2 147 483 647 (при 4 байтах).
* unsigned int: представляет положительное целое число. В зависимости от архитектуры процессора может занимать 2 байта (16 бит) или 4 байта (32 бита), и из-за этого диапазон предельных значений может меняться: от 0 до 65535 (для 2 байт), либо от 0 до 4 294 967 295 (для 4 байт).
* long: представляет целое число в диапазоне от −2 147 483 648 до 2 147 483 647. Занимает в памяти 4 байта (32 бита).
* unsigned long: представляет целое число в диапазоне от 0 до 4 294 967 295. Занимает в памяти 4 байта (32 бита).
* long long: представляет целое число в диапазоне от −9 223 372 036 854 775 808 до +9 223 372 036 854 775 807. Занимает в памяти, как правило, 8 байт (64 бита).
* unsigned long long: представляет целое число в диапазоне от 0 до 18 446 744 073 709 551 615. Занимает в памяти, как правило, 8 байт (64 бита).
* float: представляет вещественное число ординарной точности с плавающей точкой в диапазоне +/- 3.4E-38 до 3.4E+38. В памяти занимает 4 байта (32 бита)
* double: представляет вещественное число двойной точности с плавающей точкой в диапазоне +/- 1.7E-308 до 1.7E+308. В памяти занимает 8 байт (64 бита)
* long double: представляет вещественное число двойной точности с плавающей точкой не менее 8 байт (64 бит). В зависимости от размера занимаемой памяти может отличаться диапазон допустимых значений.