**Лабораторна робота №10**

**Розробка та реалізація програм з використанням бітових операцій**

**Ціль роботи**: оволодіння навичками складання програм з використанням бітових операцій

**Завдання**

**Завдання 10.1.** Визначити дію фрагмента програми

*Розв’язання*

#include<stdio.h>

#include <limits.h>

int main() {

const int N = 5;

unsigned int value = N;

size\_t len = sizeof(int) \* CHAR\_BIT; //Количество бир int-переменной - 32

const unsigned int mask = 1 << len - 1; //2^(len-1) - 2^31

printf("\nThe bitwise representation is: ");

for (int i = 0; i < len; i++) {

putchar(mask & value ? '1' : '0'); //выводим старший бит

value <<= 1; //смещаем влево (старшим битом становится предыдущий

if ((i + 1) % 8 == 0) putchar(' '); //пробел каждые 8 итераций

}

return 0;

}

Після виконання программи на екран буде виведенно:

00000000 00000000 00000000 00000101

Тобто представлення номера варианта у вдійкогому вигляді.

**Завдання 10.2.** Дано число n<32. Запишіть число 2n, тобто число, у якого n-й біт

дорівнює 1, а решта – нулі. Результат навести у 2, 8, 16 та 10 системах числення.

*Розв’язання*

1. Дано число n<32. Запишіть число 2^n, тобто число, у якого n-й біт

дорівнює 1, а решта – нулі. Результат навести у 2, 8, 16 та 10 системах числення.

1. Алгоритм розв’язання задачі:

Алгоритм розв’язання задачі можна представити у вигляді такої послідовності дій:

1. Отримати n
2. Вивести 2^n у 2, 8, 16 СЧ
3. **Текст програми**

#include <iostream>

#include <string>

#include <windows.h>

using namespace std;

int getNumber(const char label[]);

string getBin(long long n);

int main() {

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

while (1) {

int n = getNumber("Введите n: ");

long long res = long long(1) << n;

cout << dec << "pow(2, " << n << "):" << endl

<< "\t" << getBin(res) << " - в 2-ичной СК" << endl

<< "\t" << oct << res << " - в 8-ричной СК" << endl

<< "\t" << dec << res << " - в 10-ичной СК" << endl

<< "\t" << hex << res << " - в 16-ричной СК " << endl;

}

system("pause");

return 0;

}

string getBin(long long n) {

string bin;

while (n) {

bin += to\_string(n % 2);

n /= 2;

}

reverse(bin.begin(), bin.end());

return bin;

}

int getNumber(const char label[]) {

cout << label;

while (true) {

int val;

cin >> val;

if (cin.fail()) {

cin.clear();

cin.ignore(32767, '\n');

cout << "Произошла ошибка. Введите еще раз: ";

}

else {

cin.ignore(32767, '\n');

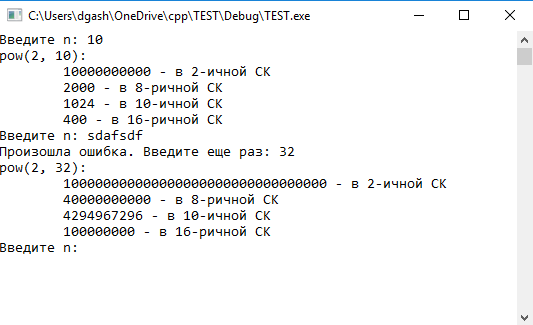
return val;

}

}

}

1. Результат виконання програми:



**Висновок:** на цій лабораторній роботі оволодів навичками складання програм для роботи зі структурами