# Міністерство освіти і науки України

Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського"

Факультет інформатики та обчислювальної техніки Кафедра інформатики та програмної інженерії

Звіт

з лабораторної роботи № 6 з дисципліни

«Алгоритми та структури даних.

Основи алгоритмізації»

Варіант 13

Виконав студент ІП-11 Калашніков Андрій Євгенович

(шифр, прізвище, ім'я, по батькові)

Перевірив Мартинова О.П.

( прізвище, ім'я, по батькові)

Київ 2021

**Лабораторна робота №6**

**Дослідження рекурсивних алгоритмів**

**Мета –** дослідити особливості роботи рекурсивних алгоритмів та набути практичних навичок їх використання під час складання програмних специфікацій підпрограм.

**Індивідуальне завдання:**

13. Перетворення значення у двійковій системі числення в десяткове значення.

**Постановка задачі**

За допомогою рекурсії, напишемо програму що буде переводити значення із двійкової системи числення в десяткову за загальними правилами.

% - знаходження остачі << - побітовий зсув

**Побудова математичної моделі**

Складемо таблицю змінних

|  |  |  |  |
| --- | --- | --- | --- |
| Змінна | Тип | Ім’я | Призначення |
| Число у двійковому вигляді | Цілий | num | Початкове дане |
| counter | Цілий | i | Початкове дане |
| Спрощення числа | Цілий | t | Проміжне дане |
| Число у десятковій системі | Цілий | p | Кінцеве дане |

**Розв’язання**

Програмні специфікації запишемо у псевдокоді та графічній формі у вигляді блок-схеми.

Крок 1. Визначимо основні дії;

Крок 2. Деталізуємо підпрограму.

**Псевдокод**

Крок 1

**Початок**

-num; i=-1

-ввід num

-обчислення десяткового числа

-вивід p

**Кінець**

Крок 2

**Початок**

-num; i=-1

-ввід num

- **Функція**

- t=0

-**Якщо** Num==0

**то Повернути 0**

**-Інакше**

**-** i=i+1

**-** t=(Num % 10) \* (1 << i) + function(Num / 10, i);

**Повернути** t

-вивід p

**Кінець**

**Блок-схема**

1. **2)**





**Код на С++ :**

#include <iostream>

using namespace std;

int function(int Num, int i);

int main()

{

int num;

int p;

int i = -1;

cout << "Enter number: ";

cin >> num;

p = function(num,i);

cout << p << endl;

}

int function(int Num, int i)

{

int t = 0;

if (Num == 0) { //umova vyhodu

return 0;

}

else

{

i++;

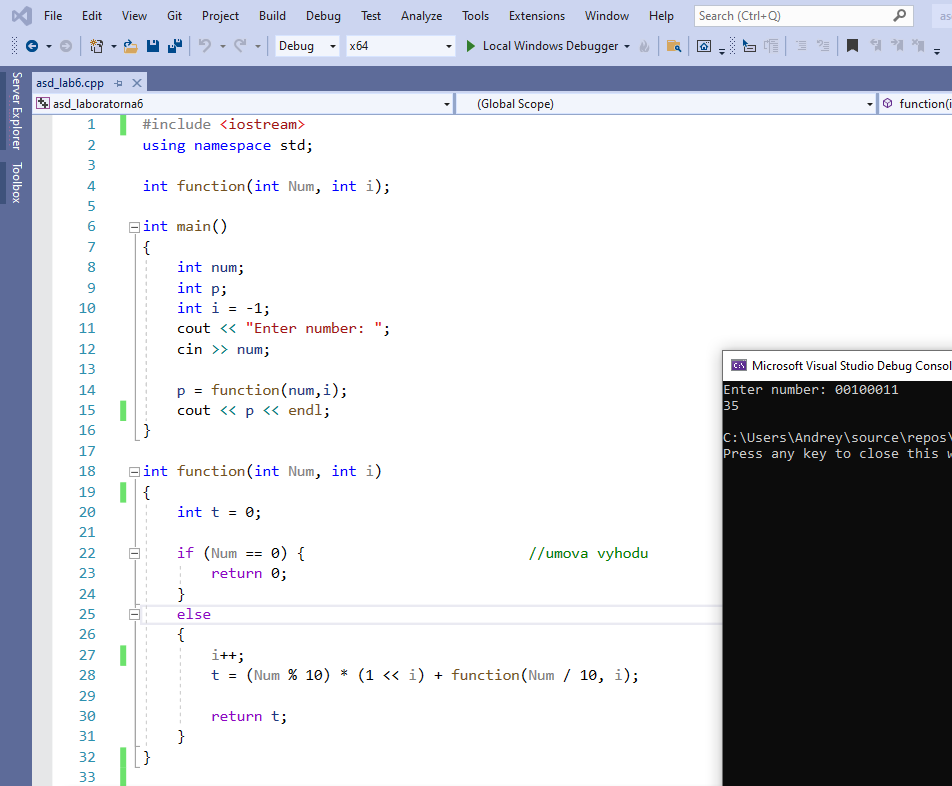
t = (Num % 10) \* (1 << i) + function(Num / 10, i);

return t;

}

}

**Скріншот результатів C++ :**

****

**Випробування**

|  |  |
| --- | --- |
| Блок | Дія |
|  | Початок |
| 1 | i=-1; |
| 2 | Ввід num=00100011 |
| 3 | i=0, num=10001 |
| 4 | i=1, num=1000 |
| 5 | i=2, num=100 |
| 6 | i=3, num=10 |
| 7 | i=4, num=1 |
| 8 | i=5, t=35, p=35 |
|  | Вивід: 35 |
|  | Кінець |

**Висновок**

Під час лабораторної роботи ми дослідили особливості роботи рекурсивних алгоритмів та набути практичних навичок їх використання під час складання програмних специфікацій підпрограм. Математична модель, блок-схеми, псевдокод, код випробовування наведені. Оскільки перевірені вручну результати розрахунку збігаються з програмними, то алгоритм правильно працює та подає результат.