

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-1.
Основи алгоритмізації»

«Дослідження лінійних алгоритмів»

Варіант 29

Виконав студент ІІІ-13 Романюк Діана Олексіївна
(шифр, прізвище, ім'я, по батькові)

Перевірів Вечерковська Анастасія Сергіївна
(прізвище, ім'я, по батькові)

Лабораторна робота 6

Дослідження рекурсивних алгоритмів

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

Варіант 29

Постановка задачі:

Перетворити натуральне число з десяткової системи числення у двійкову

Побудова математичної моделі:

Таблиця імен змінних

Змінна	Тип	Ім'я	Призначення
Задане натуральне число	Цілий	Num	Вхідні та проміжні дані

Задача:

Rec() – функція для переведення заданого числа із десяткової у двійкову систему числення.

У наступних кроках ми будемо виконувати цілочисельне да ділення з остачею над змінною **Num**, тож задаємо умову повернення рекурсивної функції у випадку коли це число буде дорівнювати 0.

Далі викликаємо цю саму функцію, але для числа (**Num \ 2**)

\ - функція цілочисельного ділення

Далі виводимо остачу від ділення **Num** на 2

Тобто **Num % 2**

% - остача від ділення на число.

Розв'язання

Крок 1. Визначимо основні дії

Крок 2. Деталізуємо дію переведення заданого числа у двійкову систему за допомогою рекурсивної функції.

Псевдокод алгоритму головної функції:

Крок 1

початок

введення Num

використання рекурсивної функції для переведення заданого числа у двійкову систему

вивід результату обчислень функції

кінець

Крок 2

початок

введення Num

Rec()

вивід обчислень Rec(Num)

кінець

Псевдокод алгоритму рекурсивної функції:

Rec(Num)

якщо Num = 0

то

повернути 0

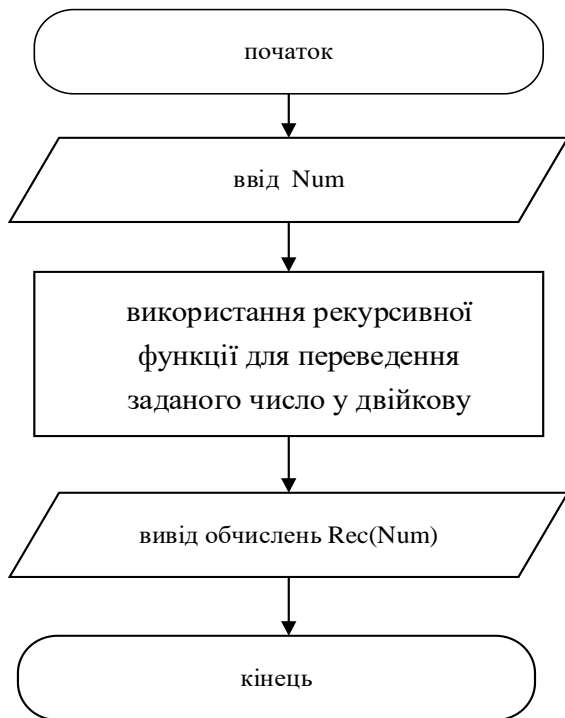
все якщо

Rec(Num / 2)

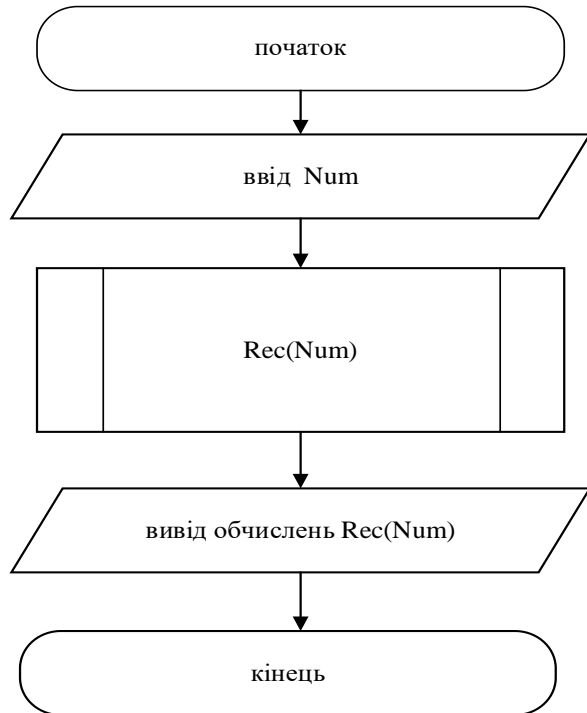
виведення Num % 2

Блок-схема головної функції:

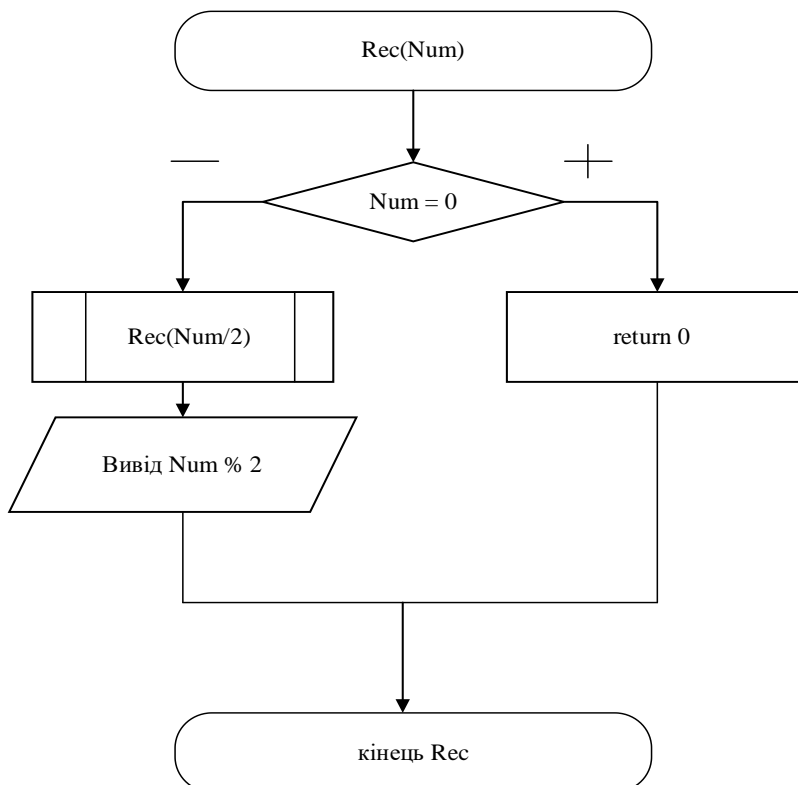
Крок 1



Крок 2



Блок-схема рекурсивної функції:



Код програми:

```
Lab6.cpp*  X
Прочие файлы (Глобальная область)
1  #include<iostream>
2  using namespace std;
3
4
5
6  int Rec(int Num) {
7      if (Num == 0)
8          return 0;
9      Rec(Num / 2);
10     cout << Num % 2;
11 }
12
13 int main() {
14     int Num;
15     cout << "Enter number:" << endl;
16     cin >> Num;
17     Rec(Num);
18 }
```

```
Консоль отладки Microsoft Visual Studio
Enter number:
6
110
C:\Users\Asus\source\repos\ConsoleApplication3\Debug\ConsoleApplication3.exe (процесс 13740) завершил работу с кодом 0.
Чтобы автоматически закрывать консоль при остановке отладки, включите параметр "Сервис" ->"Параметры" ->"Отладка" -> "Автоматически закрыть консоль при остановке отладки".
Нажмите любую клавишу, чтобы закрыть это окно...
```

```
Консоль отладки Microsoft Visual Studio
Enter number:
10
1010
C:\Users\Asus\source\repos\ConsoleApplication3\Debug\ConsoleApplication3.exe (процесс 13652) завершил работу с кодом 0.
Чтобы автоматически закрывать консоль при остановке отладки, включите параметр "Сервис" ->"Параметры" ->"Отладка" -> "Автоматически закрыть консоль при остановке отладки".
Нажмите любую клавишу, чтобы закрыть это окно...
```

Випробування алгоритму:

Блок	Дія
	Початок
1	Введення b
2	Вивід 110
	Кінець

Блок	Дія
	Початок
1	Введення 10
2	Вивід 1010
	Кінець

Висновки:

Під час виконання лабораторної роботи досліджено особливості роботи рекурсивних алгоритмів. Отримано практичні навички їх використання під час складання лінійних програмних специфікацій. Побудовано математичну модель задачі та таблицю імен змінних. Розроблено псевдокод вирішення даної математичної задачі. Умовно розбито виконання коду на кроки, а також описано його виконання за допомогою створення відповідної блок-схеми. Перевірено умовне виконання коду за допомогою випробування алгоритму.