

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 рекурсивних алгоритмів»

Варіант 29

Виконав студент

ІІІ-15 Рибалка Ілля Сергійович
(шифр, прізвище, ім'я, по батькові)

Перевірів

Всечерковська Анастасія Сергіївна
(прізвище, ім'я, по батькові)

Лабораторна робота 6

Дослідження рекурсивних алгоритмів

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

Індивідуальне завдання

Варіант 29

Перетворення натурального числа з десяткової системи числення у двійкову.

1. Постановка задачі

Перевести число десяткової системи числення у двійкову шляхом перетворення натурального числа в рекурсивній функції.

2. Побудова математичної моделі

Змінна	Тип	Ім'я	Призначення
Основна програма			
Десяткове число	Натуральне	num	Вхідні дані
Двійкове число	Натуральне	res	Вихідні дані
Переведення у двійкову систему	Підпрограма	bin	Початкові дані
Десяткове число	Натуральне, 0	num	Вхідні дані
Двійкове число	Натуральне, 0	binum	Вхідні, проміжні дані
Остача від ділення	Натуральне, 0	bit	Проміжні дані

Алгоритм переведення в двійкову систему схожий на роботу рекурсивної функції.

- Спочатку - *рекурсивний спуск*, цілочисельне ділення десяткового числа на 2 (для отримання цілої частини від ділення на 2 використаємо оператор *div*), поки не отримаємо результат 1 або 0.
- Потім - *рекурсивне повернення*, починаючи з результату записуємо остачу від ділення (для отримання остачі від ділення використаємо оператор *mod*), зліва на право.

Це і буде виконувати підпрограма *bin*.

В основній програмі вводиться десяткове число *num*, наступною дією є присвоєння змінній *res* значення підпрограми *bin*.

Вхідними даними підпрограми є *num*, змінній *binum* присвоюється значення 0.

Основи програмування – 1. Алгоритми та структури даних

- Рекурсивний спуск відбувається до того моменту, поки *num* не набуде значення меншого за 2, для цього змінна *binum* набуває значення підпрограми *bin(num div 2)*.
- Після цього починається рекурсивне повернення зі знаходженням змінної *bit* і додаванням її до *binum*10*.

Підпрограма повертає значення *binum*, що присвоюється змінній *res* в основній програмі, останньою дією є виведення *res* в консоль.

Розв'язання

Програмні специфікації запишемо у псевдокоді та графічній формі у вигляді блок-схеми.
Крок 1. Визначимо основні дії.

Крок 2. Деталізуємо крок знаходження *res* за рахунок підпрограми.

Крок 3. Визначимо основні дії підпрограми.

Крок 4. Деталізуємо дію рекурсивного спуску.

Крок 5. Деталізуємо дію знаходження *bit*.

Крок 6. Деталізуємо дію знаходження *binum*.

Псевдокод

Основна програма

крок 1

початок

Введення *num*

Знаходимо *res*

Виведення *res*

кінець

крок 2

початок

Введення *num*

res = *bin (num)*

Виведення *res*

кінець

Підпрограма

крок 3

bin (*num*)

binum = 0

Рекурсивний спуск

Знаходження *bit*

Знаходження *binnum*

повернути *binnum*

кінець **bin**

крок 4

bin (num)

binum = 0

якщо num > 1

binum = bin (num div 2)

все якщо

Знаходження bit

Знаходження binnum

повернути binnum

кінець bin

крок 5

bin (num)

binum = 0

якщо num > 1

binum = bin (num div 2)

все якщо

bit = num mod 2

Знаходження binnum

повернути binnum

кінець bin

крок 6

bin (num)

binum = 0

якщо num > 1

binum = bin (num div 2)

все якщо

bit = num mod 2

binum *= 10

binum += bit

повернути binnum

кінець bin

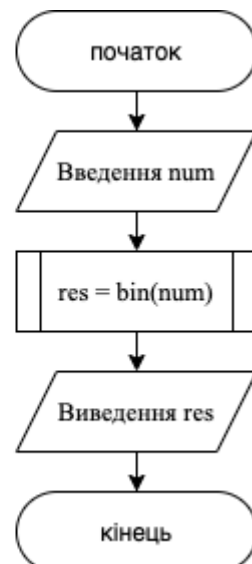
Блок-Схема

Основна програма

крок 1



крок 2

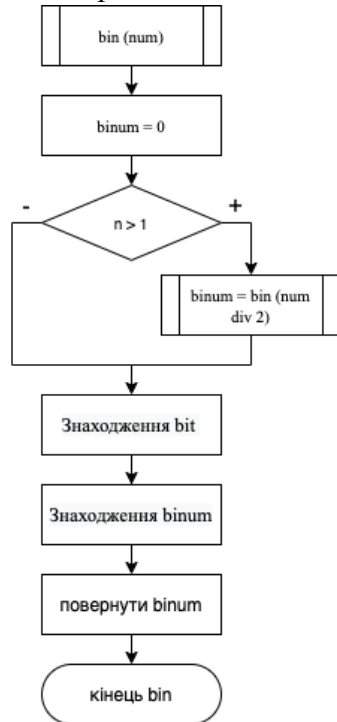


Підпрограма

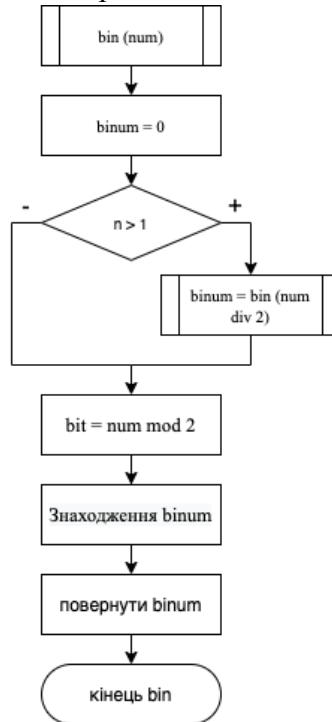
крок 3



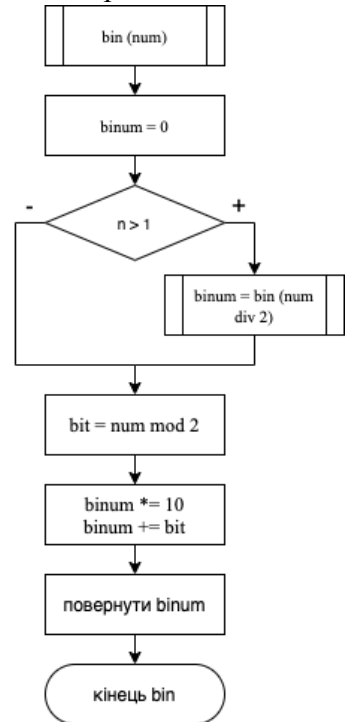
крок 4



крок 5



крок 6



Код

```

#include <iostream>

int bin(int) ;
int main()
{
    int num, res;
    std::cout << "Введіть натуральне число десяткової системи = ";
    std::cin >> num;
    res = bin(num);
    std::cout << "Число " << num << " в двійковій системі числення: " << res << std::endl;
    return 0;
}

int bin(int num)
{
    int bit, binum = 0;
    if (num > 1)
    {
        binum = bin(num / 2);
    }
    bit = num % 2;
    binum *= 10;
    binum += bit;
    return binum;
}
  
```

Тестування

Блок	Дія
	Початок
1	num = 14
2	bin(14)
3	14 > 1 == True, bin (14 div 2)
4	7 > 1 == True, bin (7 div 2)
5	3 > 1 == True, bin (3 div 2)
6	1 > 1 == False
7	bit = 1 mod 2 = 1, binum = 0*10 + 1 = 1
8	bit = 3 mod 2 = 1, binum = 1*10 + 1 = 11
9	bit = 7 mod 2 = 1, binum = 11*10 + 1 = 111
10	bit = 14 mod 2 = 0, binum = 111*10 + 0 = 1110
11	res = 1110
12	Виведення 1110
	Кінець

Висновок

Я дослідив особливості роботи рекурсивних алгоритмів та набув практичних навичок їх використання під час складання програмних специфікацій підпрограм. Було створено алгоритм переведення числа з десяткової, в двійкову систему числення. Алгоритм було протестовано на числі 14, результатом слугувало число 1110.