

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 11

Виконав студент ІП-13, Дем'янчук Олександр Петрович
(шифр, прізвище, ім'я, по батькові)

Перевірив Вєчерковська А. С.
(прізвище, ім'я, по батькові)

Лабораторна робота 6

Дослідження рекурсивних алгоритмів

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

Індивідуальне завдання

Варіант 11

Завдання

Обчислити добуток 4 елементів геометричної прогресії, що зростає: початкове значення – 5, крок – 3

1. Постановка задачі

Методом рекурсивного алгоритму маємо обрахувати добуток 4 елементів геометричної прогресії, що зростає, для таких значень: $b1 = 5$, $q = 3$, $n = 4$;

2. Математична модель

Побудуємо таблицю імен змінних:

<i>Змінна</i>	<i>Тип</i>	<i>Ім'я</i>	<i>Призначення</i>
Початкове значення	Цілий	b1	Вхідні дані
Крок прогресії	Цілий	q	Вхідні дані
Степеневий параметр	Цілий	n	Вхідні дані
Функція степені	Цілий	pow()	Проміжні дані
Кінцеве значення	Цілий	progress	Вихідні дані

Рекурсивна функція **Mult_of_progression(b1, q, n)** матиме три параметри:

b1 - перший член геометричної прогресії;

q - крок геометричної прогресії;

n - степеневий параметр прогресії

Умовою рекурсивної функції буде $n > 1$: поки виконується ця умова, призначаємо **progress** значення **$b1 * pow(q, n-1) * Mult_of_progression(b1, q, n-1)$** .

Коли ж умова $n > 1$ перестане виконуватись, тобто **n** досягне значення 1, призначаємо **progress** значення **b1**

Розв'язання

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

Крок 1. Визначимо основні дії

Крок 2. Деталізуємо дію опису рекурсивного алгоритму **Mult_of_progression(b1, q, n-1)**

Псевдокод

Крок 1

реалізація рекурсії

початок

_____ введення b1, q, n

виклик функції **Mult_of_progression(b1, q, n-1)**

виведення **Mult_of_progression(b1, q, n-1)**

кінець

Крок 2

функція **Mult_of_progression(b1, q, n-1)**

якщо $n > 1$ то

return $b1 * \text{pow}(q, n-1) * \text{Mult_of_progression}(b1, q, n-1)$

інакше

return b1

все якщо

все функція

початок

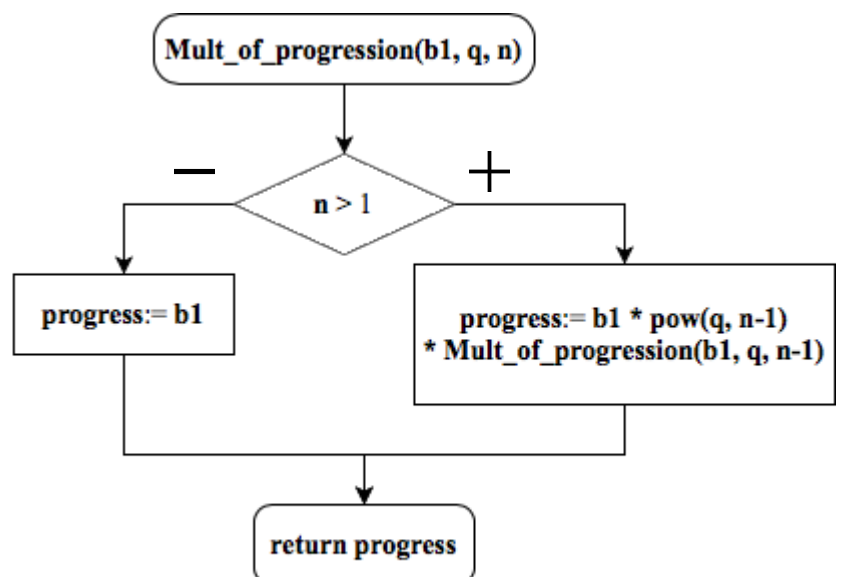
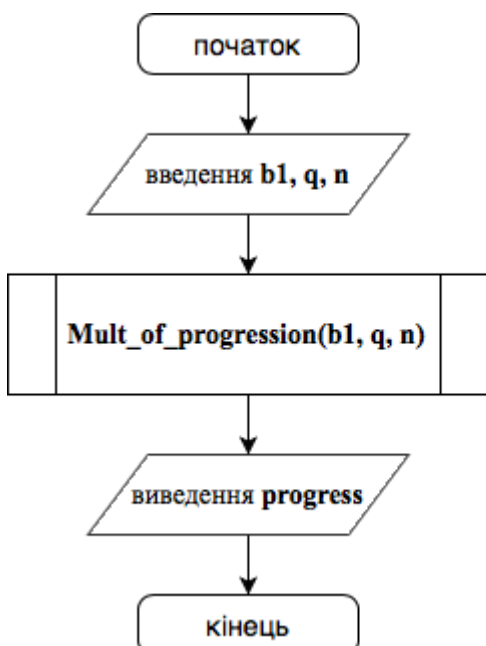
введення b1, q, n

виклик функції **Mult_of_progression(b1, q, n-1)**

виведення **Mult_of_progression(b1, q, n-1)**

кінець

Блок-схема



Тестування

Блок	Дія
	Початок
1	введення $b1 = 5, q = 3, n = 4$
2	виклик функції <code>Mult_of_progression(5, 3, 4)</code>
3	$4 > 1 = \text{true}$
4	<code>progress = 5 * pow(3, 3) * Mult_of_progression(5, 3, 3)</code> // виклик функції
5	$3 > 1 = \text{true}$
6	<code>progress = 5 * pow(3, 2) * Mult_of_progression(5, 3, 2)</code> // виклик функції
7	$2 > 1 = \text{true}$
8	<code>progress = 5 * pow(3, 1) * Mult_of_progression(5, 3, 1)</code> // виклик функції
9	$1 > 1 = \text{false}$
10	<code>progress = 5</code>
11	<code>return 5</code> // повернення до кроку 8
12	<code>return 75</code> // повернення до кроку 6
13	<code>return 3375</code> // повернення до кроку 4
14	<code>return 455625</code> // повернення до кроку 2
15	виведення <code>progress = 455625</code>
	Кінець

Програмний код

```
1  #include <iostream>
2  #include <math.h>
3  using namespace std;
4  int progress = 0;
5  int Mult_of_progression(int b1, int q, int n)
6  {
7
8      if(n>1)
9      {
10         progress = b1 * pow(q, n-1) * Mult_of_progression(b1, q, n-1);
11     }
12     else
13     {
14         progress = b1;
15     }
16     return progress;
17 }
18 int main()
19 {
20     int b1 = 5;
21     int q = 3;
22     int n = 4;
23     Mult_of_progression(b1, q, n);
24     cout << progress << endl;
25 }
```

ПРОБЛЕМЫ ВЫХОДНЫЕ ДАННЫЕ КОНСОЛЬ ОТЛАДКИ ТЕРМИНАЛ

```
MacBook-Pro-user:lab5 user$ cd "/Users/user/Documents/Лаби ОП/lab5/" && g++ tempCodeRunnerF
455625
MacBook-Pro-user:lab5 user$ █
```

Висновок

На лабораторній роботі дослідив особливості роботи рекурсивних алгоритмів, на практиці навчився складати та використовувати їх під час складання програмних специфікацій підпрограм.