

**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 22

Виконав студент ІП-14 Нікулін Павло Юрійович  
(шифр, прізвище, ім'я, по батькові)

Перевірів Мартінова Оксана Петрівна  
(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.

### Завдання

Обчислити кількість комбінацій з  $n$  різних елементів по  $m$ . Кількість комбінацій визначається формулою:

$$C_n^m = \begin{cases} 1, & \text{якщо } m = 0, n > 0 \text{ або } m = n \geq 0; \\ 0, & \text{якщо } m > n \geq 0; \\ C_{n-1}^{m-1} + C_{n-1}^m & \text{в інших випадках.} \end{cases}$$

### Розв'язання

1. **Постановка задачі:** результатом роботи має кількість комбінацій з  $n$  по  $m$  обчислених за формулою. Використаємо функція для обчислення кількості комбінація та ще одну функцію для обчислення факторіала, який потрібно буде обрахувати за формулою.
2. Побудова **математичної моделі**. Складемо таблицю імен змінних.

Змінна	Тип	Ім'я	Призначення
М елементи	Натуральне	М	Початкове дане
Н елементи	Натуральне	Н	Початкове дане
Обчислення факторіала	Ціле	ФАКТ(І)	Функція
Формальний параметр функції	Натуральне	І	Проміжне дане
Обчислення кіль-ті комбінацій	Ціле	COMB(M1, N1)	Функція
Формальний параметр функції	Натуральне	M1	Проміжне дане
Формальний параметр функції	Натуральне	N1	Проміжне дане
Кількість комбінацій з N по M	Ціле	X	Результат

*Крок 1.* Визначимо основні дії.

*Крок 2.* Дано значення числам  $n$  та  $m$ .

*Крок 3.* Опишемо функцію знаходження факторіала.

*Крок 4.* Опишемо функцію знаходження кількості комбінацій.

*Крок 5.* Знайдемо значення  $x$ .

*Псевдокод*

*крок 1*

**початок**

Дано значення числам  $m$  та  $n$

Опишемо функцію знаходження факторіала

Опишемо функцію знаходження кількості комбінацій

Знайдемо значення  $x$

**кінець**

*крок 2*

**початок**

$m = 3; n = 8;$

Опишемо функцію знаходження факторіала

Опишемо функцію знаходження кількості комбінацій

Знайдемо значення  $x$

**кінець**

*крок 3*

**початок**

$m = 3; n = 8;$

**початок**  $fact(i)$

**якщо**  $I == 0$  **то** return 1

**інакше** return  $i * fact(i - 1)$

**все якщо**

**кінець**  $fact(i)$

Опишемо функцію знаходження кількості комбінацій

Знайдемо значення  $x$

**кінець**

*крок 4*

**початок**

$m = 3; n = 8;$

**початок**  $fact(i)$

**якщо**  $I == 0$  **то** return 1

**інакше** return  $i * fact(i - 1)$

**все якщо**

**кінець**  $fact(i)$

**початок**  $comb(m1, n1)$

**якщо**  $(m1 == 0 \text{ та } n1 > 0)$  **або**  $(m1 == n1 \text{ та } n1 \geq 0)$  **то** return 1

**інакше якщо**  $m1 > n1$  **та**  $n1 \geq 0$  **то** return 0

**інакше** return  $\text{fact}(n1) / (\text{fact}(m1) * \text{fact}(n1 - m1));$

**все якщо**

**кінець**  $\text{fact}(m1, n1)$

Знайдемо значення  $x$

**кінець**

*крок 5*

**початок**

$m = 3; n = 8;$

**початок**  $\text{fact}(i)$

**якщо**  $i == 0$  **то** return 1

**інакше** return  $i * \text{fact}(i - 1)$

**все якщо**

**кінець**  $\text{fact}(i)$

**початок**  $\text{comb}(m1, n1)$

**якщо**  $(m1 == 0 \text{ та } n1 > 0)$  **або**  $(m1 == n1 \text{ та } n1 \geq 0)$  **то** return 1

**інакше якщо**  $m1 > n1 \text{ та } n1 \geq 0$  **то** return 0

**інакше** return  $\text{fact}(n1) / (\text{fact}(m1) * \text{fact}(n1 - m1));$

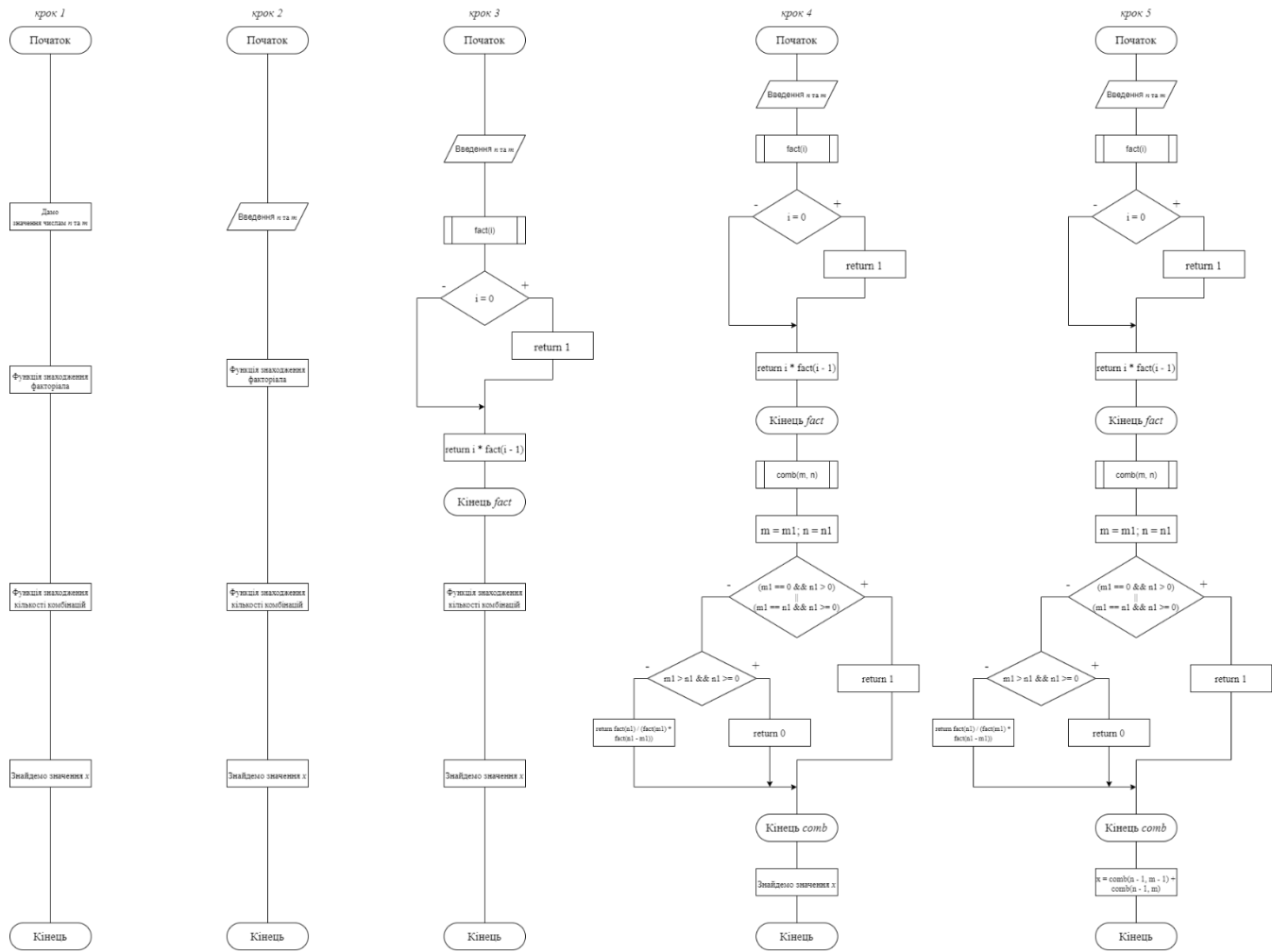
**все якщо**

**кінець**  $\text{fact}(m1, n1)$

$x = \text{comb}(n - 1, m - 1) + \text{comb}(n - 1, m);$

**кінець**

## Блок-схема



## Код програми

```
1  #include <iostream>
2  using namespace std;
3
4  /*Функція розрахування факторіалу*/
5  int fact(int i)
6  {
7      if (i == 0)
8      {
9          return 1;
10     }
11     else
12     {
13         return i * fact(i - 1);
14     }
15 }
16
17 /*Функція розрахування комбінацій*/
18 int comb(int n1, int m1)
19 {
20     if ((m1 == 0 && n1 > 0) || (m1 == n1 && n1 >= 0))
21     {
22         return 1;
23     }
24     else if (m1 > n1 && n1 >= 0)
25     {
26         return 0;
27     }
28     else
29     {
30         return fact(n1) / (fact(m1) * fact(n1 - m1));
31     }
32 }
33
34 int main()
35 {
36     int m, n, x;
37
38     cout << "input m: ";
39     cin >> m;
40     cout << "input n: ";
41     cin >> n;
42
43     x = comb(n - 1, m - 1) + comb(n - 1, m);
44
45     cout << x;
46 }
```

## Тестування програми

```
input m: 5
input n: 13
1287
```

## Математичне доведення

$$C_n^m = C_{n-1}^{m-1} + C_{n-1}^m = C_{12}^4 + C_{12}^5 = \frac{12!}{4!(12-4)!} + \frac{12!}{5!(12-5)!} = 495 + 792 = 1287$$

## **Висновок**

Під час виконання лабораторної роботи було досліджено особливості роботи рекурсивних алгоритмів та набуто практичних навичок їх використання під час складання програмних специфікацій підпрограм. Було створено дві функції: для обчислення факторіала та для знаходження кількості комбінацій. Обидві функції є рекурсивними. Також в роботі присутні оператори *if*. Програма написана на основі алгоритму працює коректно.