

Міністерство освіти і науки України
Вінницький національний технічний університет
Факультет інформаційних технологій та комп'ютерної інженерії
Кафедра ПЗ

Лабораторна робота №4, варіант № 19
з дисципліни «Основи програмування»

Виконала: ст. 1ПІ-25Б

Перевірив: доцент

Семенов В.О. .

Решетнік О.О.

Тема: Функції та рекурсії

Мета: Створення простих та рекурсивних функцій, передача параметрів, повернення значень.

Завдання 19. Написати функцію для обчислення значення

$$S(n) = \sum_{i=1}^n \frac{(2i)!}{(n+1)!}$$

Обчислити з її допомогою значення $S(n)$ для n від 12 до 24 із кроком 4.

Декомпозиція

1. Обчислення факторіалу

Створити функцію, яка обчислює факторіал числа k .

Факторіал обчислюється рекурсивно (для будь-якого числа $k > 1$, результат дорівнює k помножити на факторіал числа $k-1$).

2. Обчислення суми $S(n)$

Створити функцію, яка обчислює суму рекурсивно:

Базовий випадок: якщо поточний індекс більше n , повертається 0.

Інакше додається поточний доданок $(2i)!/(n+1)!$ і викликається функція для наступного індексу.

3. Перебір значень n

Визначити діапазон n від 12 до 24 з кроком 4.

Для кожного значення n викликати функцію обчислення суми.

4. Вивід результатів

Відобразити для кожного n обчислене значення суми $S(n)$.

Код програми показано на лістингу 1 а блок-схему на рисунку 1.

Лістинг 1 – Програма для знаходження суми

```
#include <stdio.h>

long double fact(int k)
{
    if (k <= 1) return 1.0;
    return k * fact(k - 1);
}
```

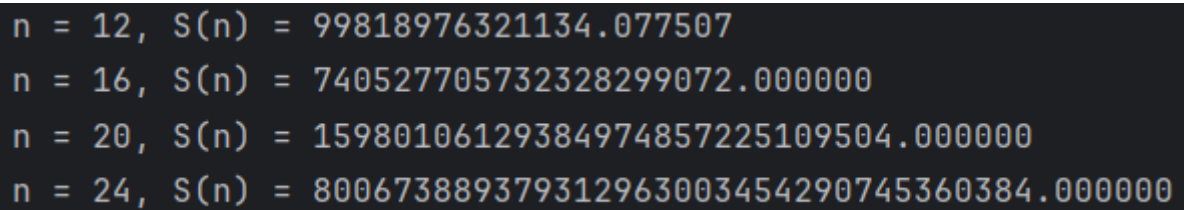
Продовження лістингу 1

```
}

long double recursive(int n, int i)
{
    if (i > n) return 0.0;
    return fact(2 * i) / fact(n + 1) + recursive(n, i + 1);
}

int main(void)
{
    for (int n = 12; n <= 24; n += 4)
    {
        long double S = recursive(n, 1);
        printf("n = %d, S(n) = %.6Lf\n", n, S);
    }
    return 0;
}
```

Приклади роботи програми можна побачити на рисунку 1



```
n = 12, S(n) = 99818976321134.077507
n = 16, S(n) = 740527705732328299072.000000
n = 20, S(n) = 15980106129384974857225109504.000000
n = 24, S(n) = 800673889379312963003454290745360384.000000
```

Рисунок 1 – приклад роботи програми

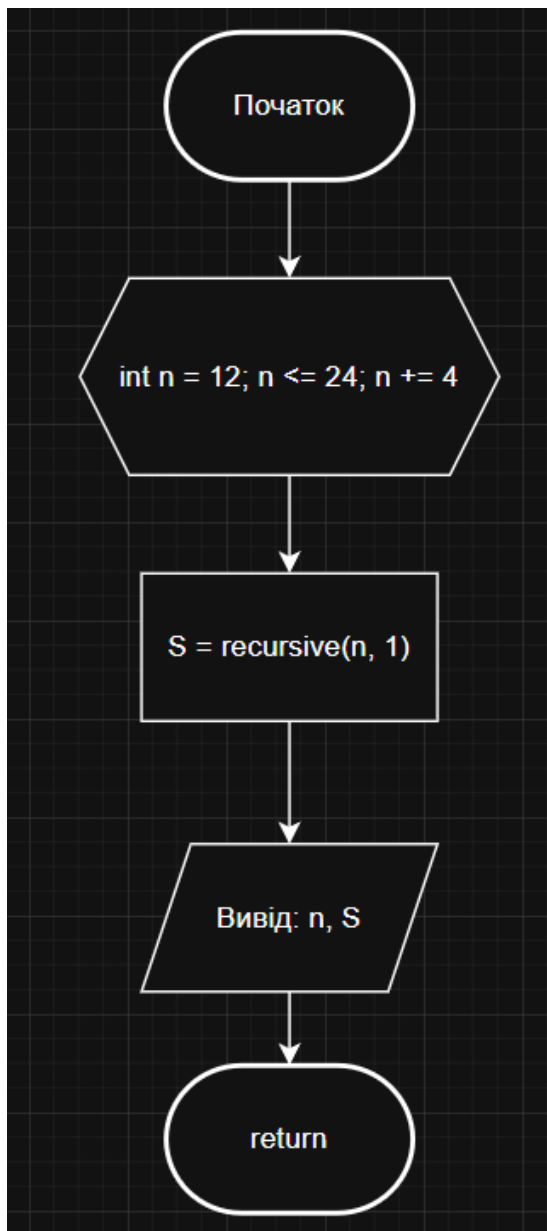


Рисунок 4 – Блок-схема main

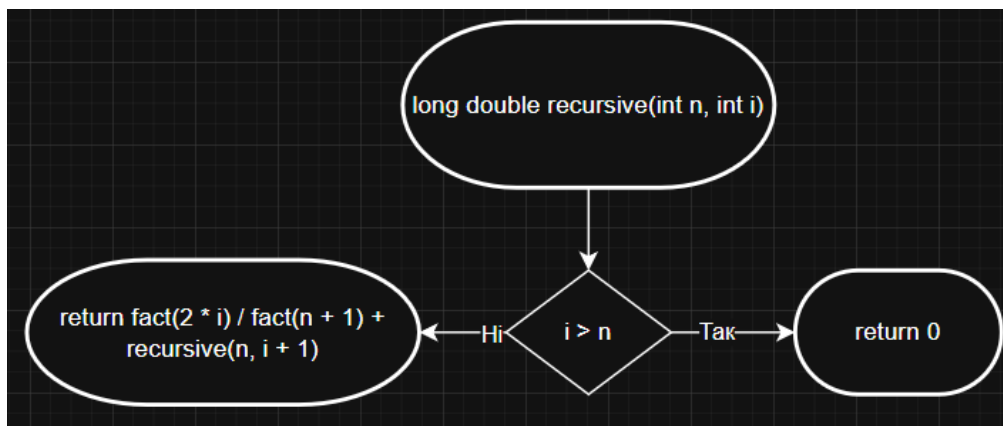


Рисунок 5 – Блок-схема recursive

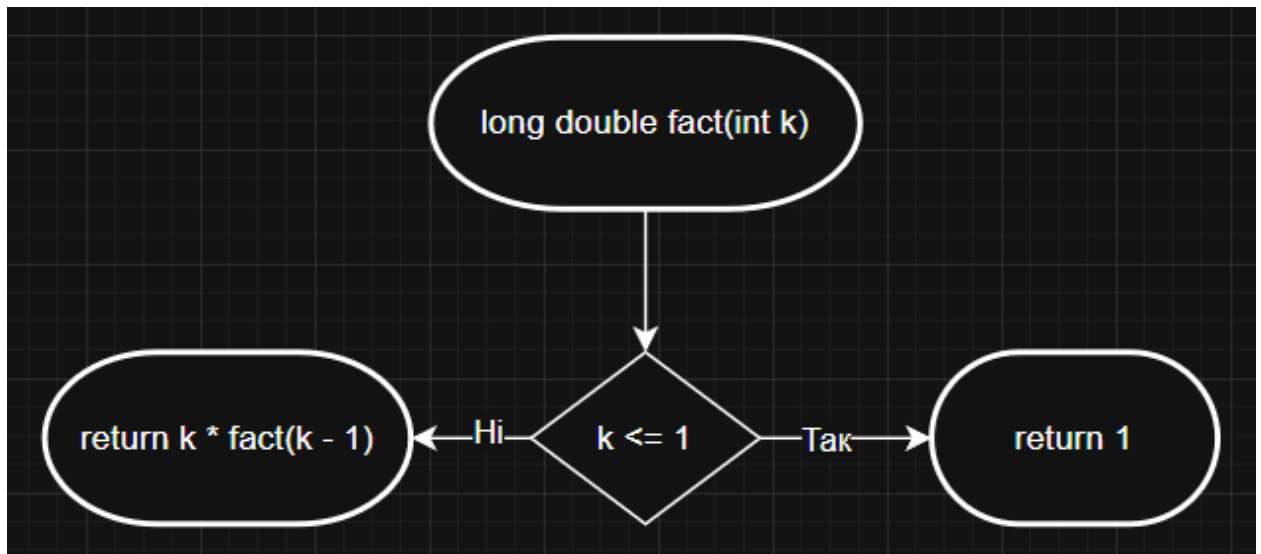


Рисунок 6 – Блок-схема fact

Контрольні запитання:

4. Будова функції

Функція складається з оголошення (назва, тип, параметри), тіла (команди для виконання завдання) і оператора `return` (якщо функція повертає значення).

5. Способи повернення результату в C

Через `return` (повертає значення функції).

Через параметри за адресою (передати вказівник на змінну, яку функція змінює).

6. Місце розташування функцій у файлі

Перед функцією `main()` (щоб виклик був відомий компілятору).

Після `main()`, якщо перед цим зроблено `forward declaration` функції.

Висновок:

У ході лабораторної роботи було відпрацьовано створення простих та рекурсивних функцій, вивчено передачу параметрів і способи повернення значень із функцій. Було реалізовано рекурсивну функцію для обчислення суми з факторіалами та відпрацьовано її виклик для різних значень `n`. Робота дозволила закріпити навички розробки функцій і рекурсії в мові Cі.