ЗВІТ

про виконання лабораторної роботи No < 5.1 >

«Обчислення сум та добутків за допомогою рекурсії»

з дисципліни

«Алгоритмізація та програмування»

Студента групи КН-106Б

Онишківа Остапа Володимировича

**Мета:** Навчитися використовувати рекурсивні функції.

**Умова завдання:** За допомогою рекурсивних функцій обчислити значення суми чи добутку. В одній програмі слід вивести результати звертання до п’яти функцій, які реалізують наступні способи:

1) обчислення здійснюються на рекурсивному спуску, значення параметра в рекурсивних викликах – спадає;

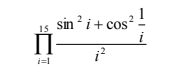
2) обчислення здійснюються на рекурсивному спуску, значення параметра в рекурсивних викликах – зростає;

3) обчислення здійснюються на рекурсивному підйомі, значення параметра в рекурсивних викликах – спадає;

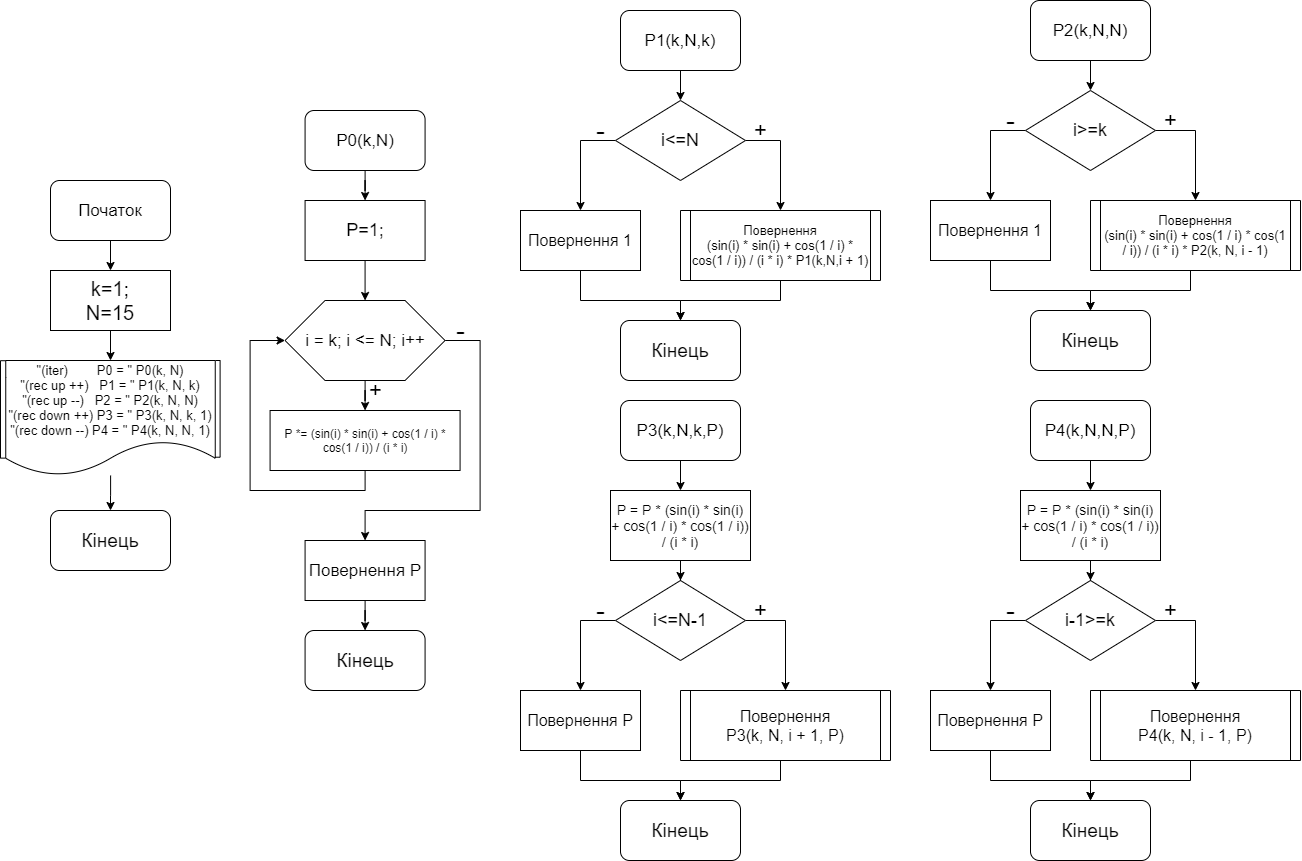
4) обчислення здійснюються на рекурсивному підйомі, значення параметра в рекурсивних викликах – зростає;

5) обчислення здійснюється ітераційним способом (для контролю правильності рекурсивних способів).

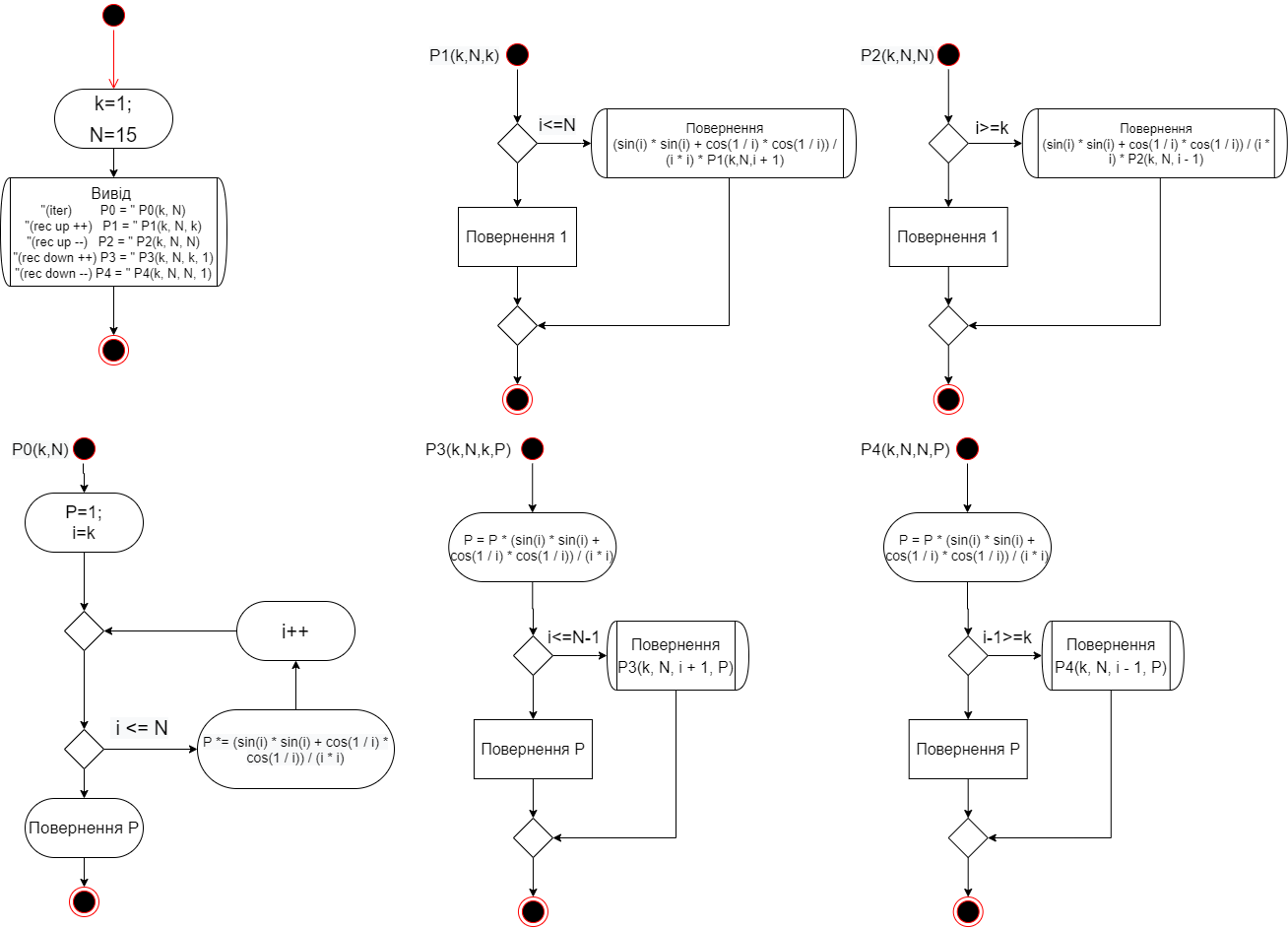
Всі п’ять результатів мають збігатися.

****

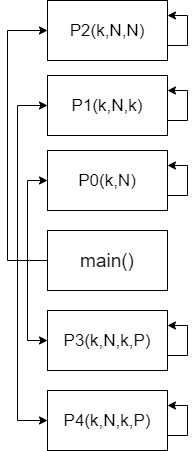
**Блок-схема алгоритму:**

****

**UML-діаграма алгоритму:**

****

**Структурна схема:**

****

**Текст програми:**

// Lab\_5.4.cpp

// < Онишківа Остапа >

// Лабораторна робота No 5.4

// Обчислення сум та добутків за допомогою рекурсії.

// Варіант 5

#include<iostream>

#include<cmath>

using namespace std;

double P0(int k, int N)

{

double P = 1;

for (int i = k; i <= N; i++)

{

P \*= (sin(i) \* sin(i) + cos(1 / i) \* cos(1 / i)) / (i \* i);

}

return P;

}

double P1(int k, int N,int i)

{

if (i<=N)

{

return (sin(i) \* sin(i) + cos(1 / i) \* cos(1 / i)) / (i \* i) \* P1(k,N,i + 1);

}

else

{

return 1;

}

}

double P2(int k, int N, int i)

{

if (i>=k)

{

return (sin(i) \* sin(i) + cos(1 / i) \* cos(1 / i)) / (i \* i) \* P2(k, N, i - 1);

}

else

{

return 1;

}

}

double P3(int k, int N, int i, double P)

{

P = P \* (sin(i) \* sin(i) + cos(1 / i) \* cos(1 / i)) / (i \* i);

if (i<=N-1)

{

return P3(k, N, i + 1, P);

}

else

{

return P;

}

}

double P4(int k, int N, int i, double P)

{

P = P \* (sin(i) \* sin(i) + cos(1 / i) \* cos(1 / i)) / (i \* i);

if (i-1>=k)

{

return P4(k, N, i - 1, P);

}

else

{

return P;

}

}

int main()

{

int k, N;

k = 1;

N = 15;

cout << "(iter) P0 = " << P0(k, N) << endl;

cout << "(rec up ++) P1 = " << P1(k, N, k) << endl;

cout << "(rec up --) P2 = " << P2(k, N, N) << endl;

cout << "(rec down ++) P3 = " << P3(k, N, k, 1) << endl;

cout << "(rec down --) P4 = " << P4(k, N, N, 1) << endl;

}

**Результат виконання:**



**Посилання на git-репозиторій з проектом:**

**Текст програми unit-тесту:**

**Результат unit-тесту:**

**Висновки:** на цій лабораторній роботі я навчився користуватися функціями, малювати структурні схеми програми та робити unit-тести.