**1. Мета роботи:**

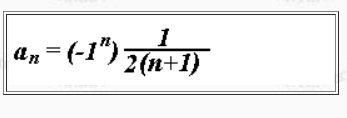
Метою лабораторної роботи є отримання практичних навичок в роботі з операторами циклу мови C.

**2. Теми для попереднього опрацювання:**

Оператори циклу мови C.

**3. Завдання для виконання**

Для ряду, члени якого обчислюються за формулою, що відповідає Вашим індивідуальним завданням, підрахувати суму членів ряду з точністю до 0.000001 і суму перших 10 членів ряду. Якщо Ви вважаєте це за необхідне, можете спростити або перетворити вираз.



**Загальний метод вирішення**

Очевидно, що процес підрахунку суми членів ряду повинен бути ітераційним: слід повторювати обчислення по одній і тій же формулі при значеннях n = 0, 1, 2, .... У кожній ітерації циклу слід виконувати обчислення за заданою формулою для поточного значення n, тобто підраховувати черговий член ряду. Отримане значення слід додавати до змінної, яка представляє суму. Ця змінна в кожній ітерації буде містити в собі суму всіх вже оброблених членів ряду, отже, її початкове значення (коли жоден член ще не оброблений) має бути 0. Після обчислення суми при значенні n = 9 слід вивести значення суми - це один з результатів програми відповідно до завдання (береться значення 9, тому що перший член ряду обчислюється при n = 0, таким чином, дев'ятий - при n = 9). Після обчислення кожного члена ряду (але до збільшення його значення до суми) слід порівняти отримане значення з заданим межею точності. Через те, що значення члена ряду може бути як позитивним, так і негативним, при порівнянні слід використовувати абсолютне значення. Якщо абсолютне значення члена ряду не перевищує межі точності, слід закінчити обчислення: вийти з циклу надрукувати значення суми і завершити програму.

**Код:**

#include <iostream>

#include <math.h>

#include <stdio.h>

using namespace std;

int main(void) {

setlocale(LC\_ALL, "RUS");

long n; /\* параметр ряда \*/

double dbln; /\* параметр ряда в форме с плавающей точкой \*/

double sum = 0; /\* сумма членов ряда \*/

double term; /\* значение текущего члена \*/

const double eps = 0.000001; /\* предел точности \*/

long k2 = 1; /\* pow(2,n)\*/

short k1 = 1; /\* pow(-1,n)\*/

/\* основной цикл; в модификациях вычисляются

следующие значения pow(2,n) и pow(-1,n)\*/

for (n = 0; ; n++, k1 = -k1) {

/\* преобразование n в форму с плавающей точкой \*/

dbln = n;

/\* вычисление очередного члена \*/

term = k1 \* (1) / (2 \* (dbln + 1));

/\* проверка достижения предела точности \*/

if (fabs(term) >= eps)

/\* если не достигнут - накопление суммы \*/

sum += term;

/\* если достигнут - выход из цикла \*/

else break;

/\* если 10 членов - вывод суммы \*/

if (n == 9)

printf("Сумма 10 членов ряда = %10.7lf\n", sum);

}

/\* конец основного цикла \*/

/\* вывод окончательной суммы \*/

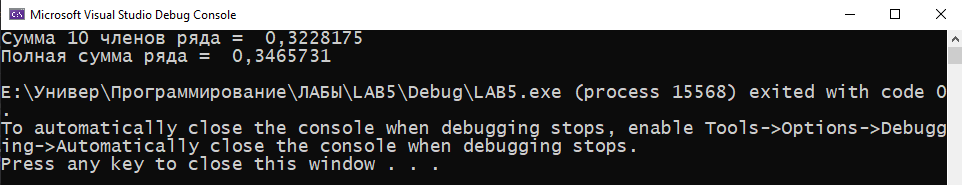
printf("Полная сумма ряда = %10.7lf\n", sum);

return 0;

} /\* конец программы \*/

**Результати роботи програми**

При роботі програми на екран були виведені такі результати:

**Висновки:**

При виконанні лабораторної роботи вивчені питання:

складання циклічних алгоритмів;

програмної реалізації циклічних алгоритмів;

налагодження програм, що містять циклічні алгоритми.