МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

* + 1. федеральное государственное автономное образовательное учреждение высшего образования

«САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ АЭРОКОСМИЧЕСКОГО ПРИБОРОСТРОЕНИЯ»

(ГУАП)

КАФЕДРА ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМ И СЕТЕЙ

Преподаватель

канд. техн. наук, доцент Л.Н. Бариков

Отчёт

по лабораторной работе №3

по дисциплине ОСНОВЫ ПРОГРАММИРОВАНИЯ

на тему: «Операторы цикла: заданное число повторений»

Работу выполнила

студентка гр. 4141 А. В. Мазанова

Санкт-Петербург

2022

**Вариант 23**

**Цель лабораторной работы:** изучение концепций и освоение технологии процедурного программирования, приобретение навыков процедурного программирования на языке C/C++ циклических вычислений.

**Задание на программирование:** используя технологию процедурного программирования разработать программу решения индивидуальной задачи тремя видами циклических управляющих структур: Цикл - Пока (с предусловием), Цикл - До (с постусловием), Цикл - Для (с параметром).

**Ход выполнения работы:**

1. **Получила индивидуальное задание:**

Для введенного с клавиатуры значения *m* вычислить *S*:

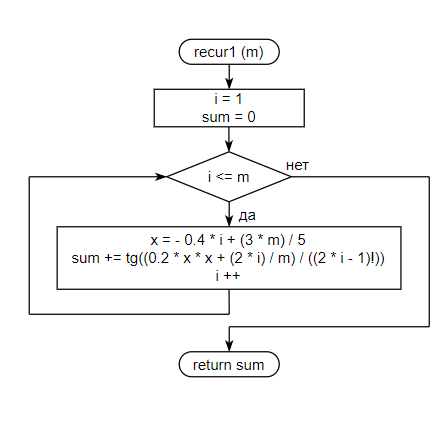


1. **Математическая модель:**

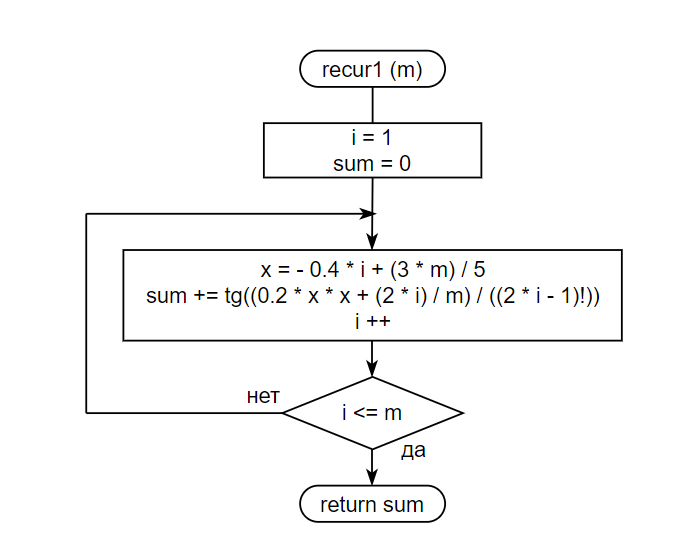
С клавиатуры вводятся значение m.  
Задается начальный номер слагаемого (*i = 1*). Обнуляется начальное значение суммы.  
Вычисляется значение знаменателя аргумента тангенса *(denom = (2×i – 1)!):* начальное его значение до цикла примем за *(1)*, то есть при *i = 1*, далее воспользуемся формулой *n! = n × (n – 1) × (n – 2)!* и будем считать его, как *(2×i – 1) × ((2×i – 1) – 1) × denom* (полученном при предыдущей итерации).  
Вычисляется значение x по заданной формуле *x = - 0.4×i + (3 × m) / 5*Вычисляется величина очередного слагаемого, и на эту величину увеличивается текущее значение суммы.  
Увеличивается на 1 значение *i*.  
Для нового значения параметра *i* вычисляется значение *denom*, значение *х,* величина очередного слагаемого, и на эту величину увеличивается текущее значение суммы  
Действия повторяются до превышения параметром *i* значения *m*

*Примечание:* при *i = 1* будем отдельно считать 1-ое слагаемое, т.к. при подстановке в рекуррентную формулу вычисления факториала множитель *((2×i – 1) – 1) = (2×i – 2)* становится равным нулю, что препятствует дальнейшим вычислениям. С учетом этого все циклы будут начинаться с   
*i = 2*.

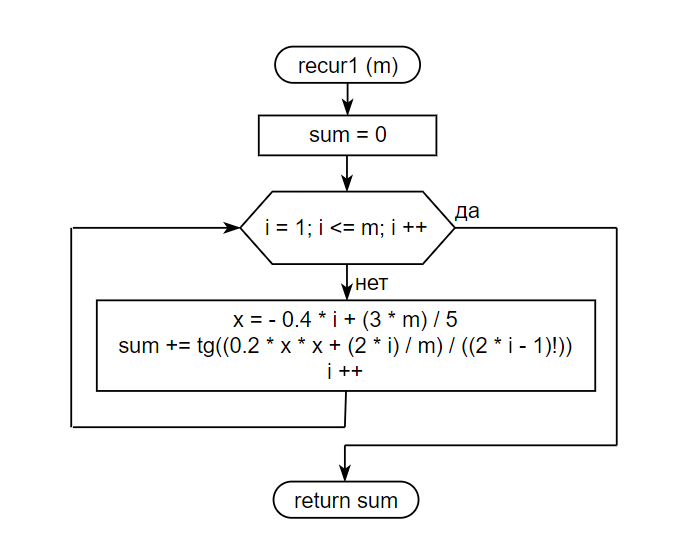
1. **Составила схему алгоритма:**
2. решение с использованием цикла ***while*** (функция *recur1*)



1. решение с использованием цикла ***do…while*** (функция *recur2*)



1. решение с использование цикла ***for*** (функция *recur3*)



1. **Составила программу на языке *С/C++***

//Для введенного с клавиатуры значения m вычислить

//значение суммы заданного вида, если значение x вычисляется

//по формуле x = - 0.4\*i + ((3 \* m) / 5).

#include <iostream>

#include <cmath>

using namespace std;

double recur1(int m);

double recur2(int m);

double recur3(int m);

int main()

{

int m; //конечное значение номера слагаемого

int var; //выбор действия

double rec1, rec2, rec3; // Результаты решения задачи

setlocale(LC\_ALL, "Russian");

for (;;)

{ //Выбор действия

cout << "\n Вид действия:" << endl;

cout << " 1 - Вычисление суммы по рекуррентной формуле" << endl;

cout << " 2 - Завершение программы" << endl;

cout << " Введите вид действия -> ";

cin >> var;

switch (var)

{

case 1: //Ввод исходных данных

cout << " Введите m -> ";

cin >> m;

rec1 = recur1(m);

rec2 = recur2(m);

rec3 = recur3(m);

//Вывод результата

cout.precision(5); //число знаков после запятой при выводе

cout << " Для цикла WHILE результат = " << rec1 << endl;

cout << " Для цикла DO..WHILE результат = " << rec2 << endl;

cout << " Для цикла FOR результат = " << rec3 << endl;

break;

default: return 1;

}

}

}

double recur1(int m) {

int i = 2; //начальное значение номера слагаемого

double denom = 1, //значение знаменателя аргумента тангенса

sum = 0, //исходное значение суммы

x = (-0.4 \* 1 + ((3. \* m) / 5))\*1.;

double sl1 = tan((0.2 \* x \* x + (2. \* 1) / 1) / denom); // отдело посчитанное 1 слагаемое

while (i <= m) {

denom \*= (2. \* i - 1) \* (2 \* i - 2);

x = (-0.4 \* i + ((3 \* m) / 5));

sum += tan((0.2 \* x \* x + (2. \* i) / m) / denom);

i++;

}

sum += sl1;

return sum;

}

double recur2(int m) {

int i = 2; //начальное значение номера слагаемого

double denom = 1, //значение знаменателя аргумента тангенса

sum = 0, //исходное значение суммы

x = (-0.4 \* 1 + ((3. \* m) / 5))\*1.;

double sl1 = tan((0.2 \* x \* x + (2. \* 1) / 1) / denom)\*1.; // отдельно посчитанное 1 слагаемое

do {

if (m == 1) {

break;

}

denom \*= (2. \* i - 1) \* (2 \* i - 2);

x = (-0.4 \* i + ((3 \* m) / 5));

sum += tan((0.2 \* x \* x + (2. \* i) / m) / denom);

i++;

} while (i <= m);

sum += sl1;

return sum;

}

double recur3(int m) {

double denom = 1, //значение знаменателя аргумента тангенса

sum = 0, //исходное значение суммы

x = (-0.4 \* 1 + ((3. \* m) / 5));

double sl1 = tan((0.2 \* x \* x + (2. \* 1) / 1) / denom); // отдело посчитанное 1 слагаемое

for (int i = 2; i <= m; i++) {

denom \*= (2. \* i - 1) \* (2 \* i - 2);

x = (-0.4 \* i + ((3 \* m) / 5));

sum += tan((0.2 \* x \* x + (2. \* i) / m) / denom);

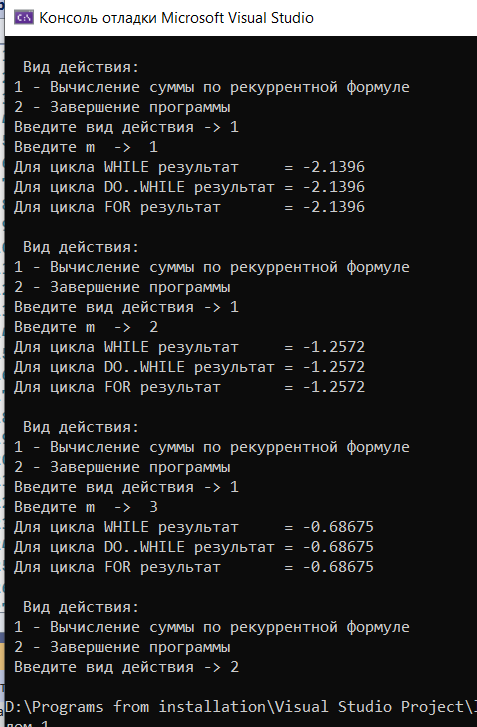
}

sum += sl1;

return sum;

}

1. **Проверила работу программы на языке *С/C++*:**

****

**Вывод:** я изучила концепции и освоила технологии процедурного программирования, приобрела навыки процедурного программирования на языке *С/С++* циклических вычислений