

ФГБОУ ВО “Чувашский государственный университет им. И. Н. Ульянова”

Факультет: ИВТ

Кафедра: Вычислительной техники

Предмет: Объектно-ориентированное программирование

Лабораторная работа №2

**Базовые конструкции структурного
программирования**

Выполнил: студент группы ИВТ-41-20

Галкин Дмитрий

Проверил: доцент Обломов И.А.

Теория

Следование – это конструкция, предоставляющая собой последовательность двух или более операторов, обеспечивающих выполнение программы «сверху вниз».

Ветвление – задает выполнение одного или другого оператора в зависимости от выполнения какого-либо условия (предиката).

Цикл – задает многократное выполнение оператора или последовательности операторов.

Условный оператор if – используется для разветвления вычислительного процесса два альтернативных направлений: if(expr) operator_1; else operator_2;

Оператор switch – используется для разветвления процесса на несколько направлений.

```
switch(expr)
{
    case конст_выражение_1: список_операторов_1;
    case конст_выражение_2: список_операторов_2;
    .....
    case конст_выражение_N: список_операторов_N;

    default: операторы;
}
```

Цикл с предусловием(while) - while(expr) оператор;

Цикл с постусловием(do while) - do оператор while (expr);

Цикл с параметром for - for(инициализация; выражение; модификация) оператор;

Индивидуальное задание (Вариант 7)

Формула – $\text{actgx} = \frac{\pi}{2} + \sum \frac{(-1)^{n+1}}{(2n+1)x^{2n+1}} = \frac{\pi}{2} - \frac{2}{x} + \frac{1}{3x^3} - \frac{1}{5x^5} + \dots x > 1$

STX – начальное значение интервала

ETX – конечное значение интервала

dx – шаг

eps – точность вычисления

```
#include <iostream>
#include <cmath>
#include <iomanip>
using namespace std;
```

```
int count;
```

```
double arctg (double x, double e) {
    double y = M_PI / 2, a = 1;
    int n = 0;
    count = 0;
    double localX;
    do {
        localX = exp(log(x) * (2 * n + 1));
        // localX = pow(x, 2 * n + 1);
        if(count % 2 == 0) {
            a = -1 / ((2 * n + 1) * localX);
        } else {
            a = 1 / ((2 * n + 1) * localX);
        }
        y += a;
        count++;
    }
```

```

        n++;
    } while(fabs(a) > e);
    return y;
}

int main() {

    system("chcp 65001");
    double STX, ENX;
    double dx;
    double e;

    cout << "Введите начальное значение интервала STX: "; cin >> STX;
    cout << "Введите конечное значение интервала ENX: "; cin >> ENX;
    cout << "Введите шаг dx: "; cin >> dx;
    cout << "Введите погрешность eps: "; cin >> e;

    cout << '|' << setw(3) << 'x' << setw(3) << '|' << setw(12) << "arctg(x)"
    << setw(5) << '|' << setw(12) << "atan(x)" << setw(5) << '|' << setw(3) << 'n' << setw(3) << '|' << endl;

    for(double x = STX; x <= ENX; x += dx) {
        double y = arctg(x, e);
        cout << '|' << setw(3) << x << setw(3) << '|' << setw(12) << y
            << setw(5) << '|' << setw(12) << atan(x) << setw(5) << '|' << setw(3) << count << setw(3) << '|' <<
endl;
    }

}

```

Вывод: Я практическим путем научился пользоваться базовыми конструкциями структурного программирования.