Министерство науки и высшего образования Российской Федерации Федеральное государственное бюджетное образовательное учреждение высшего образования «Тульский государственный университет»

КАФЕДРА ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ

**ПОДПРОГРАММЫ В ЯЗЫКЕ ПРОГРАММИРОВАНИЯ СИ**

отчет о лабораторной работе № 3

по дисциплине

*ТЕХНОЛОГИИ И МЕТОДЫ ПРОГРАММИРОВАНИЯ*

***ВАРИАНТ 13***

|  |  |  |
| --- | --- | --- |
| Выполнила: | ст. гр. 230711 | Павлова В.С. |
| Проверил: | асс. каф. ИБ | Курбаков М.Ю. |

Тула, 2023 г.

**ЦЕЛЬ И ЗАДАЧА РАБОТЫ**

**Цель:** научиться использовать функции на языке программирования Си.

**Задача:** в данной работе требуется написать программу, демонстрирующую использование изученных принципов.

**ЗАДАНИЕ НА РАБОТУ**

**Задание по теме «Функции»:**

№ 13. Задана последовательность из **N** вещественных чисел. Определить, образуют ли они возрастающую последовательность. Если последовательность не возрастает, то упорядочить ее по возрастанию.

**Задание по теме «Рекурсия»**

№ 13. Найти количество цифр заданного числа.

# **СХЕМА ПРОГРАММЫ ДЛЯ ЗАДАНИЯ ПО ТЕМЕ «ФУНКЦИИ»**

Схема алгоритма программы, предназначенной для обработки последовательности длины N, представлена на рисунке 1.

  
Рисунок 1 – Схема алгоритма программы, предназначенной для обработки последовательности длины N

# **ТЕКСТ ПРОГРАММЫ ДЛЯ ЗАДАНИЯ ПО ТЕМЕ «ФУНКЦИИ»**

Текст программы на языке программирования С/С++, предназначенной для обработки последовательности длины представлена в листинге 1.

## **Листинг 1. Текст программы**

#include <iostream>

#include <vector>

#include <algorithm>

using namespace std;

bool isIncreasing(vector<double> seq)

{

bool isIncreasing = true;

for (int i = 1; i < seq.size(); i++) {

if (seq[i] <= seq[i - 1]) {

isIncreasing = false;

break;

}

}

if (isIncreasing)

{

cout << "Последовательность возрастающая\n";

## **Листинг 1. Текст программы (продолжение)**

return true;

}

else {

sort(seq.begin(), seq.end());

cout << "Последовательность упорядочена по возрастанию: ";

for (auto elem : seq) {

cout << elem << " ";

} cout << endl;

return false;

}

}

int main() {

setlocale(LC\_ALL, "russian");

int n;

cout << "Введите количество элементов в последовательности: "; cin >> n;

vector<double> seq(n);

cout << "Введите элементы последовательности: ";

for (int i = 0; i < n; i++) {

cin >> seq[i];

}

isIncreasing(seq);

return 0;}

**ИНСТРУКЦИЯ ПРОГРАММИСТА ДЛЯ ЗАДАНИЯ ПО ТЕМЕ «ФУНКЦИИ»**

Структуры данных, используемые в программе, приведены в таблице 1.

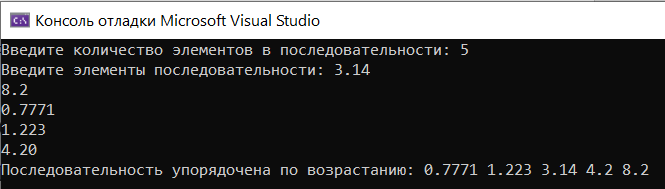
Таблица 1 – Структуры данных в программе

|  |  |  |
| --- | --- | --- |
| **Имя** | **Тип** | **Предназначение** |
| n | int | Длина последовательности |
| seq | vector<double> | Массив для хранения последовательности |
| isIncreasing | bool | Флаг |

# **ДЕМОНСТРАЦИОННЫЙ ПРИМЕР ДЛЯ ЗАДАНИЯ ПО ТЕМЕ «ФУНКЦИИ»**

Для невозрастающей последовательности длиной N = 5*,* имеющей вид *{3.14, 8.2, 0.7771, 1.223, 4.20}* результат упорядочивания по возрастанию будет следующим: {*0.7771, 1.223, 3.14, 4.2, 8,2}.*

Результат работы программы приведён на рисунке 2.

  
Рисунок 2 – Пример работы программы

# **СХЕМА ПРОГРАММЫ ДЛЯ ЗАДАНИЯ ПО ТЕМЕ «РЕКУРСИЯ»**

Схема алгоритма программы, предназначенной для поиска количества цифр заданного числа представлена на рисунке 3.



Рисунок 3 – Схема алгоритма программы, предназначенной для поиска количества цифр заданного числа

# **ТЕКСТ ПРОГРАММЫ ДЛЯ ЗАДАНИЯ ПО ТЕМЕ «РЕКУРСИЯ»**

Текст программы на языке программирования С/С++, предназначенной для поиска количества цифр заданного числа представлена в листинге 2.

## **Листинг 2. Текст программы**

#include <iostream>

using namespace std;

int checkNumbers(int num)

{

int count = 0;

while (num != 0){

## **Листинг 2. Текст программы (продолжение)**

num /= 10;

count++;

}

return count;

}

int main()

{

setlocale(LC\_ALL, "RUSSIAN");

int num;

cout << "Введите число: "; cin >> num;

cout << "Количество цифр в числе: " << checkNumbers(num) << endl;

return 0;}

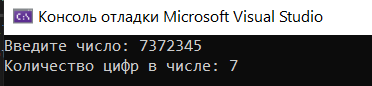
**ИНСТРУКЦИЯ ПРОГРАММИСТА ДЛЯ ЗАДАНИЯ ПО ТЕМЕ «РЕКУРСИЯ»**

Структуры данных, используемые в программе, приведены в таблице 2.

Таблица 2 – Структуры данных в программе

|  |  |  |
| --- | --- | --- |
| **Имя** | **Тип** | **Предназначение** |
| num | int | Число |
| count | int | Счётчик цифр |

# **ДЕМОНСТРАЦИОННЫЙ ПРИМЕР ДЛЯ ЗАДАНИЯ ПО ТЕМЕ «РЕКУРСИЯ»**

  
Рисунок 4 – Пример работы программы

# **ВЫВОДЫ**

В ходе выполнения работы я ознакомилась с подпрограммами на языке С/С++. Подпрограмма — именованная, логически законченная группа операторов языка, которую можно вызвать для выполнения любое количество раз из различных мест программы. В языке C++ подпрограммы реализованы в виде функций. Функция — это поименованный набор описаний и операторов, выполняющих определённую задачу. Функция может принимать параметры и возвращать значение.