Министерство науки и высшего образования Российской Федерации

Пермский национальный исследовательский политехнический университет

Электротехнический факультет

Кафедра Информационные технологии и автоматизированные системы

Лабораторная работа №7

по дисциплине

«Программирование»

Вариант 8

Выполнил: студент группы Асу15-бз

Чиков Дмитрий Александрович

Проверил: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

г. Пермь – 2020

**Цель работы:**

1) Получить практические навыки работы с функциями;

2) получить практические навыки работы с шаблонами функций;

3) получить практические навыки работы с указателями функций.

**Задачи работы:**

1. Написать функцию с умалчиваемыми параметрами в соответствии с вариантом, продемонстрировать различные способы вызова функции:

* с параметрами заданными явно,
* с опущенными параметрами
* часть параметров задана явно, а часть опущена.

2. Написать функцию с переменным числом параметров в соответствии с вариантом, продемонстрировать вызов функции с различным числом параметров.

3. Написать перегруженные функции в соответствии с вариантом. Написать демонстрационную программу для вызова этих функций.

4. Написать шаблон функций вместо перегруженных функций из задания 3. Написать демонстрационную программу для вызова этих функций. списка параметров

5. Решить уравнение указанным в варианте методом. Уравнение передать в функцию как параметр с помощью указателя.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 8 | Печать названия экзамена, даты экзамена и среднего балла | Минимальный из элементов в списке параметров, стоящих на нечетных местах | Поиск заданного элемента в массиве | Метод Ньютона    Отрезок, содержащий корень: [0;0,85]  Точное значение: 0,2624 |

Для выполнения работы были реализованы следующие функции:

exam\* init

int Und(int k, ...)

void search(int i,exam a[], const char\* object)

char\* search()

template<class T, class T1, class T2> T Search(T object, T1 i, T2 a)

float testf(float x)

float root(fptr f, float a, float b, float e)

А так же основная функция main(), из которой происходит вызов всех остальных функций. Листинг приведен в приложении А.

Результаты тестов отображены на рисунках 1-6.

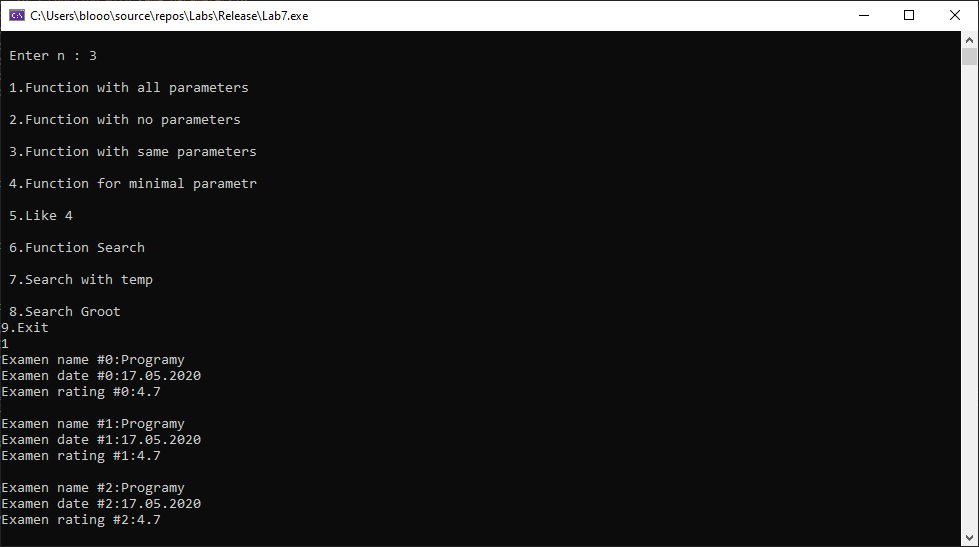


Рисунок Генерация массива с функцией со всеми параметрами

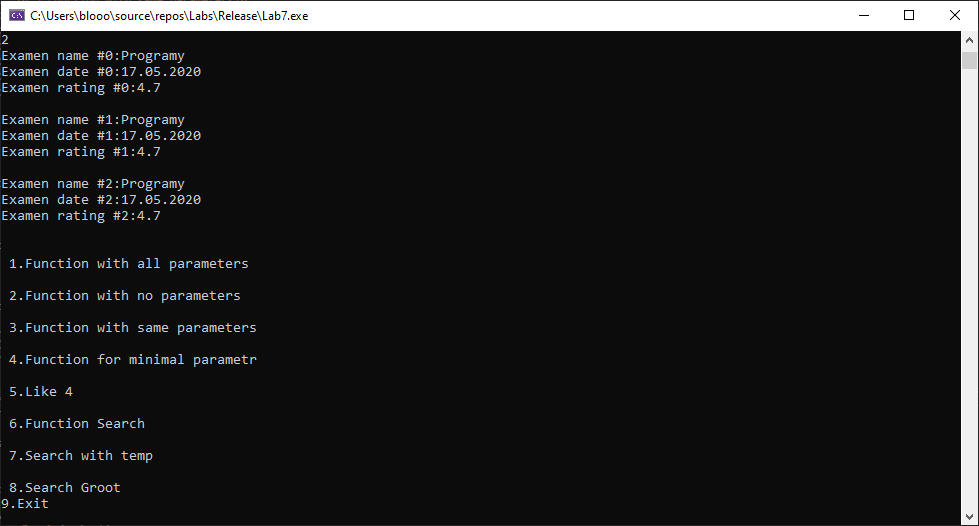


Рисунок Генерация массива с функцией без передаваемых параметров

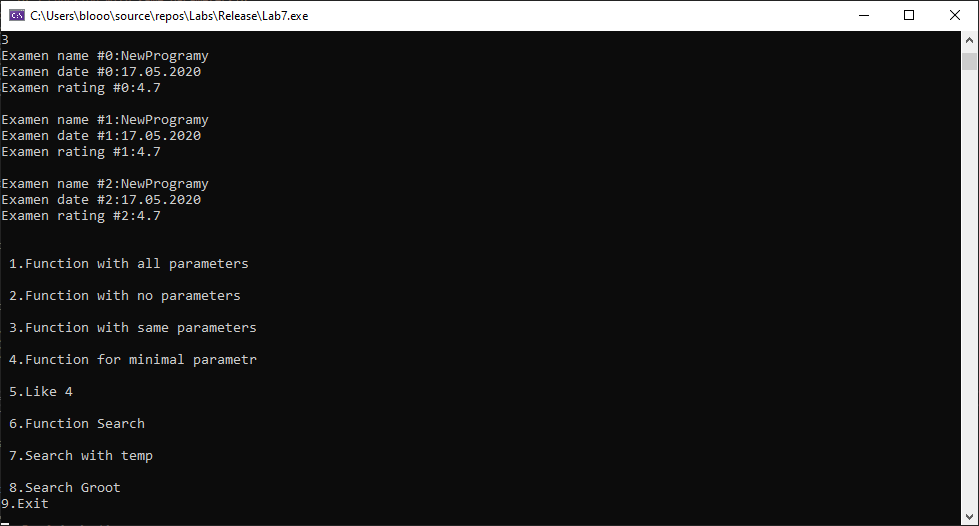


Рисунок Генерация массива с функцией с частично передаваемыми параметрами

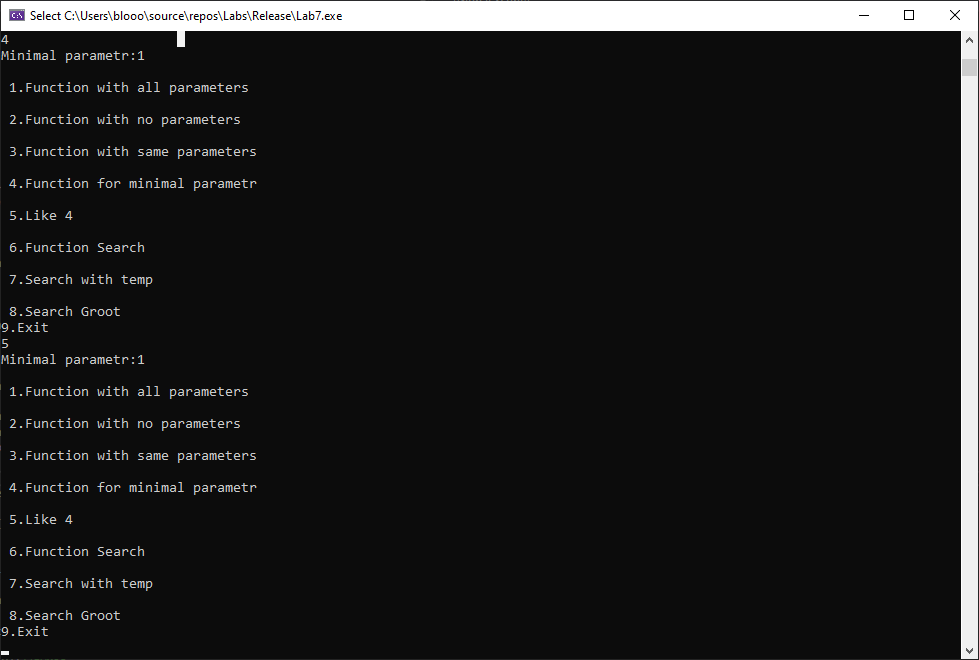


Рисунок Нахождение минимального параметра функции с помощью функции и шаблона

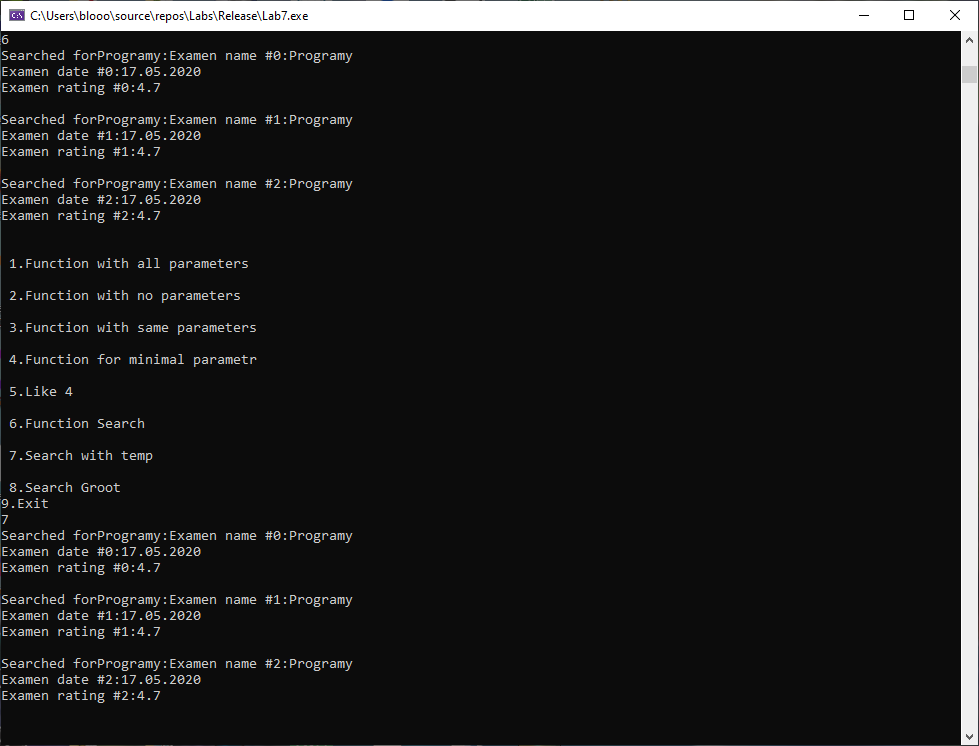


Рисунок Функция поиска и шаблон функции поиска

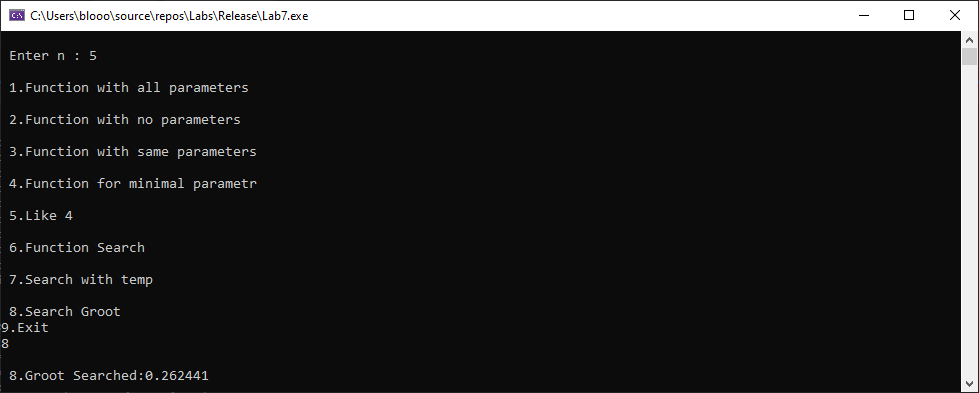


Рисунок Поиск корня

**Приложение А**

#include <cstdlib>

#include <iostream>

#include <ctime>

#include <string>

#include <stdio.h>

struct exam

{

char\* name;

char\* date;

float rating;

};

exam\* a{};

typedef float(\*fptr)(float);//тип-указатель на функцию уравнения

exam\* init(int i, exam a[] , float rating = 4.7, const char\* name = "Programy", const char\* date = "17.05.2020") {

a[i].name = new char[strlen(name) + 1];

strcpy\_s(a[i].name, 20, name);

a[i].date = new char[strlen(date) + 1];

strcpy\_s(a[i].date, 20, date);

a[i].rating= rating;

std::cout << "Examen name #" << i <<":"<< a[i].name << "\n";

std::cout << "Examen date #" << i << ":" << a[i].date << "\n";

std::cout << "Examen rating #" << i << ":" << a[i].rating << "\n\n";

return a;

}

int Und(int k, ...)

{

int\* p = &k;

int min = \*p;

for (int i = 1; p[i] != 0; i++) {

if (i % 2 == 0)

{

if (min > p[i]) { min = p[i]; }

}

}

std::cout << "Minimal parametr:" << min << "\n";

return min;

}

void search(int i,exam a[], const char\* object)

{

if (strcmp(a[i].name,object)==0)

{

std::cout << "Searched for" << object << ":";

std::cout << "Examen name #" << i << ":" << a[i].name << "\n";

std::cout << "Examen date #" << i << ":" << a[i].date << "\n";

std::cout << "Examen rating #" << i << ":" << a[i].rating << "\n\n";

}

}

char\* search() {

std::cout << "Mind of Soul dind't searcher" << "\n";

char\* string = new char[50];

strcpy\_s(string, 50, "Mind of Soul dind't searcher");

return string;

}

template<class T, class T1, class T2>

T Search(T object, T1 i, T2 a)

{

if (strcmp(a[i].name, object) == 0)

{

std::cout << "Searched for" << object << ":";

std::cout << "Examen name #" << i << ":" << a[i].name << "\n";

std::cout << "Examen date #" << i << ":" << a[i].date << "\n";

std::cout << "Examen rating #" << i << ":" << a[i].rating << "\n\n";

}

return 0;

}

float testf(float x)

{

return x - 1 / (3 + sin(3.6 \* x));

}

float root(fptr f, float a, float b, float e)

{

std::srand(std::time(nullptr));

float x0 = (b + a) / 2;

float x = x0,df;

int n = 20;

df = (f(x + e) - f(x)) / e;

for (int i = 1; i <= n; i++)

x = x - f(x) / df;

return x;

}

int main()

{

int k, n;

float x;

std::cout << "\n Enter n : ";

std::cin >> n;

exam\* a = new exam[n];

x = root(testf, 0, 0.85, 0.0001);

do

{

std::cout << "\n 1.Function with all parameters\n";

std::cout << "\n 2.Function with no parameters\n";

std::cout << "\n 3.Function with same parameters\n";

std::cout << "\n 4.Function for minimal parametr\n";

std::cout << "\n 5.Like 4\n";

std::cout << "\n 6.Function Search\n";

std::cout << "\n 7.Search with temp\n";

std::cout << "\n 8.Search Groot\n";

std::cout << "9.Exit\n";

std::cin >> k;

switch (k)

{

case 1: for (int i = 0; i < n; i++)

{

a = init(i, a,4.7, "Programy", "17.05.2020");

} break;

case 2: for (int i = 0; i < n; i++)

{

a = init(i, a);

} break;

case 3: for (int i = 0; i < n; i++)

{

a = init(i, a, 4.7,"NewProgramy");

} break;

case 4: Und(5,6,3,6,1,68,0); break;

case 5: Und(5, 6, 3, 6, 1, 68,92,46,4,64,13,169,4,964,32,1,9684, 0); break;

case 6: for (int i = 0; i < n; i++)

{

if (a != NULL) {

search(i, a, "Programy");

}

else { search(); }

} break;

case 7:

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

{

Search("Programy", i, a);

} break;

case 8: std::cout << "\n 8.Groot Searched:"<< x; break;

}

} while (k != 9);//выход

delete[] a;

}