**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ**

### Федеральное государственное автономное образовательное учреждение высшего образования

**«Санкт-Петербургский государственный университет  
аэрокосмического приборостроения»**

КАФЕДРА № 43

ОТЧЕТ   
ЗАЩИЩЕН С ОЦЕНКОЙ

ПРЕПОДАВАТЕЛЬ

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| старший преподаватель |  |  |  | Шумова Е. О. |
| должность, уч. степень, звание |  | подпись, дата |  | инициалы, фамилия |

|  |
| --- |
| ОТЧЕТ О ЛАБОРАТОРНОЙ РАБОТЕ №6 |
| «Обработка исключительных ситуаций» |
| по курсу: объектно-ориентированное программирование |
|  |

РАБОТУ ВЫПОЛНИЛ

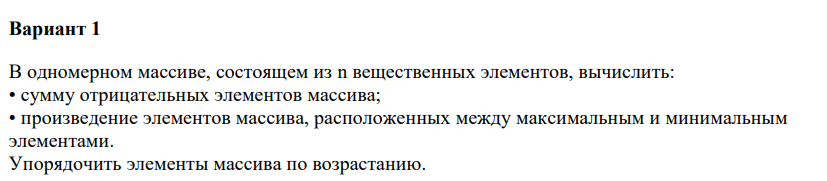
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| СТУДЕНТ ГР. № | 4134К |  |  |  | Иванов И.В. |
|  |  |  | подпись, дата |  | инициалы, фамилия |

Санкт-Петербург 2022

**Цель работы**

Изучить принципы построения консольных приложений, применив на практике знания базовых синтаксических конструкций языка C++ и объектно-ориентированного программирования.

**Задание**



**Код**

#include <iostream>

#include <string>

using namespace std;

class My\_exp

{

int id{};

std::string mes{};

public:

My\_exp();

My\_exp(int, std::string);

std::string getMes();

int getId();

};

My\_exp::My\_exp()

{

this->id = 0;

this->mes = "неизвестная ошибка\n";

}

My\_exp::My\_exp(int id, std::string mes)

{

this->id = id;

this->mes = mes;

}

std::string My\_exp::getMes()

{

return this->mes;

}

int My\_exp::getId()

{

return this->id;

}

class Array

{

int n;

double\* arr;

public:

Array(int);

~Array();

void show();

void sum();

void mult();

void sort();

};

Array::Array(int n)

{

this->n = n;

arr = new double[n];

double a;

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

{

std::cout << "введите " << i << " элемент массива "<<endl;

std::cin >> a;

arr[i] = a;

}

}

Array::~Array()

{

delete[] arr;

}

void Array::show()

{

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

{

std::cout << "array [" << i << "] = " << arr[i] << std::endl;

}

}

void Array::sum()

{

int sum;

sum = 0;

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

{

if (arr[i] < 0) {

sum = sum + arr[i];

}

else {

continue;

}

}

if (sum != 0)

{

std::cout << "Сумма отрицательных элементов " << sum << std::endl;

}

else

{

throw My\_exp(1, "Нет отрицательных элементов");

}

}

void Array::mult()

{

int max;

max = -10000;

int min;

min = 10000;

int x, y;

int mult;

mult = 1;

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

{

if (arr[i] > max) {

max = arr[i];

x = i;

}

if (arr[i] < min) {

min = arr[i];

y = i;

}

}

if(x<y)

{

throw My\_exp(2, "Начальная координата больше чем конечная");

}

if (x == y)

{

throw exception("Все элементы равны");

}

if (y < x) {

for (int i = y; i <x; i++)

{

mult=mult\* arr[i];

}

std::cout << "произведение элементов массива, расположенных между максимальным и минимальным элементами " << mult << std::endl;

}

}

void Array::sort()

{

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

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

if (arr[j] > arr[j + 1]) {

int temp = arr[j];

arr[j] = arr[j + 1];

arr[j + 1] = temp;

}

}

int main()

{

setlocale(LC\_ALL, "Russian");

system("color F0");

try

{

Array arr(5);

arr.sum();

arr.mult();

arr.sort();

arr.show();

}

catch (My\_exp& ex)

{

std::cout << "код ошибки : " << ex.getId() << std::endl;

std::cout << "сообщение: " << ex.getMes() << std::endl;

}

catch (std::exception& ex)

{

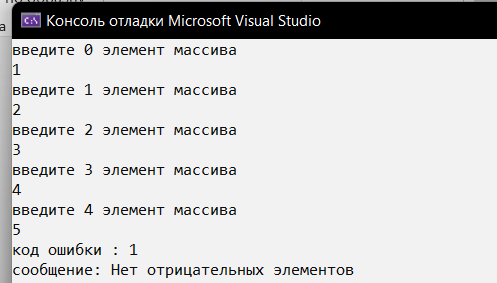
std::cout << "Error message: " << ex.what() << std::endl;

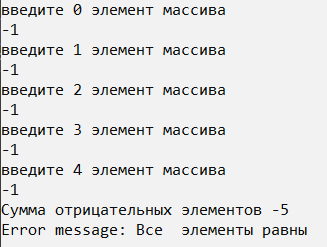
}

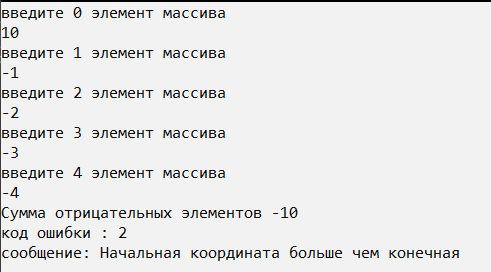
return 0;

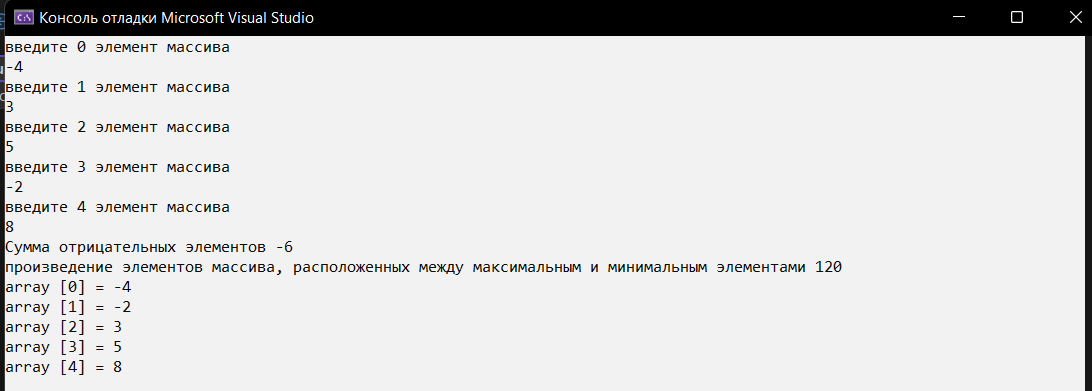
}

**Пример**









**Выводы**

Изучил принципы построения консольных приложений, применив на практике знания базовых синтаксических конструкций языка C++ и объектно-ориентированного программирования.