МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ  
федеральное государственное автономное образовательное учреждение высшего образования

«САНКТ-ПЕТЕРБУГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ  
АЭРОКОСМИЧЕСКОГО ПРИБОРОСТРОЕНИЯ»

КАФЕДРА № 43

ОТЧЕТ   
ЗАЩИЩЕН С ОЦЕНКОЙ

ПРЕПОДАВАТЕЛЬ

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

|  |
| --- |
| ОТЧЕТ О ЛАБОРАТОРНОЙ РАБОТЕ № 5 |
| «Обработка исключительных ситуаций» |
| по курсу: Объектно-ориентированное программирование |
|  |
|  |

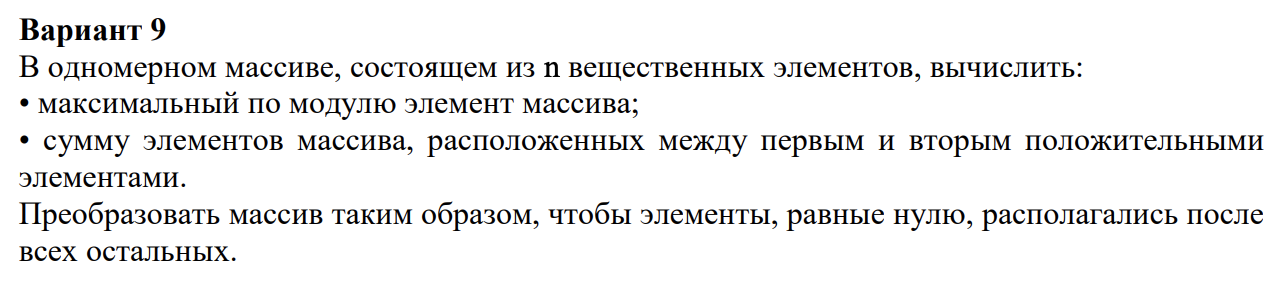
РАБОТУ ВЫПОЛНИЛ

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| СТУДЕНТ ГР. № | 4134К |  |  |  | Опарин С.Н. |
|  |  |  | подпись, дата |  | инициалы, фамилия |

Санкт-Петербург 2022

1. **Задание**

**Вариант №9:**



1. **Листинг программы, реализующей алгоритм:**

#include <iostream>

#include <string>

#include <exception>

using namespace std;

class My\_Error{

private:

string identificator;

string message;

public:

My\_Error();

My\_Error(string, string);

string getErrorIdentificator();

string getMessage();

};

My\_Error::My\_Error(){

identificator = "not defined";

message = "not defined";

}

My\_Error::My\_Error(string str, string mes){

identificator = str;

message = mes;

}

string My\_Error::getErrorIdentificator(){

return identificator;

}

string My\_Error::getMessage(){

return message;

}

class Massive{

private:

int size;

int\* mas;

public:

Massive(int);

~Massive();

void show();

void maxElement();

void amountBetweenTwoPos();

void zerosToTheEnd();

};

Massive::Massive(int size){

this->size = size;

mas = new int[size];

for (int i = 0; i < this->size; i++){

cout << "Элемент №" << i << " = ";

string read;

getline(cin, read);

mas[i] = atoi(read.c\_str());

}

}

Massive::~Massive(){

delete[] mas;

}

void Massive::show(){

cout << endl;

cout << "Массив" << endl;

for (int i = 0; i < this->size; i++)

{

cout << "Элемент №" << i << " = " << mas[i] << endl;

}

}

void Massive::maxElement(){

int masMax = 0;

for (int i = 0; i < this->size; i++){

if (abs(mas[i]) > masMax){

masMax = abs(mas[i]); // присваиваем значение нового большего элемента

}

}

if (masMax > 0){

cout << endl << "Максимальный по модулю элемент = " << masMax << endl;

}

else{

throw My\_Error("3", "no max Element");

}

}

void Massive::amountBetweenTwoPos(){

int position\_first\_positive\_element = -1;

int position\_second\_positive\_element = -1;

for (int i = 0; i < this->size; i++){

if (mas[i] > 0 && position\_first\_positive\_element == -1){

position\_first\_positive\_element = i;

}

else if (mas[i] > 0 && position\_first\_positive\_element != -1)

{

position\_second\_positive\_element = i;

break;

}

}

int amount = 0;

if (position\_first\_positive\_element != -1 && position\_second\_positive\_element != -1){

for (int i = ++position\_first\_positive\_element; i < position\_second\_positive\_element; i++){

amount += mas[i];

}

}

if (amount != 0){

cout << endl << "Сумма элементов между положительными = " << amount << endl;

}

else

{

throw My\_Error("1", "notPositiveElementOrNotBetweenSumElement");

}

}

void Massive::zerosToTheEnd(){

for (int i = 0; i < this->size - 1; i++){

for (int j = 0; j < this->size - i - 1; j++){

if (abs(mas[j]) < abs(mas[j + 1])){

int temp = mas[j];

mas[j] = mas[j + 1];

mas[j + 1] = temp;

}

}

}

}

int main(){

try{

Massive mas(5);

mas.show();

mas.maxElement();

mas.amountBetweenTwoPos();

mas.zerosToTheEnd();

cout << endl << "Перемещение нулей в конец массива" << endl;

mas.show();

}

catch (My\_Error& ex){

cout << endl << "Код ошибки: " << ex.getErrorIdentificator() << endl;

cout << "Ошибка: " << ex.getMessage() << endl;

}

catch (exception& ex)

{

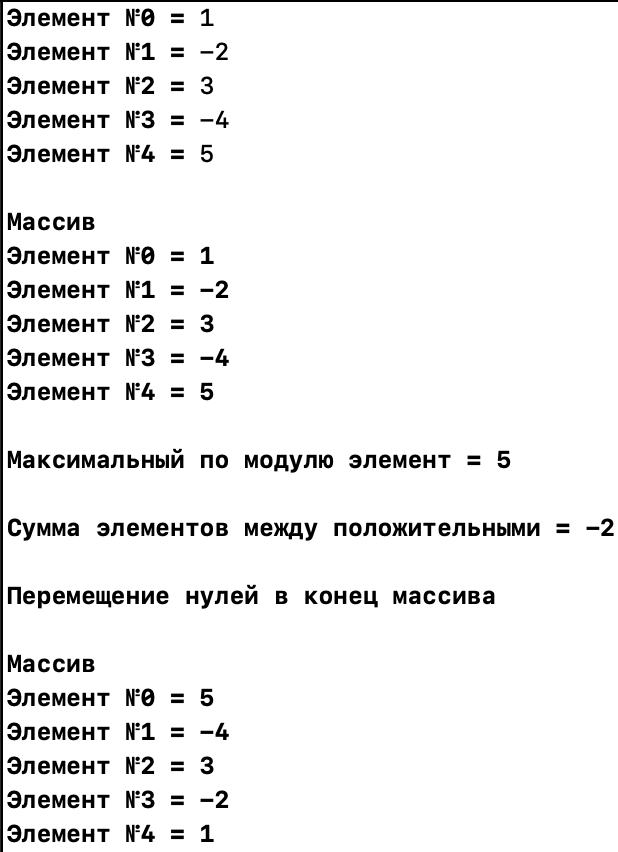
cout << endl << "Ошибка: " << ex.what() << endl;

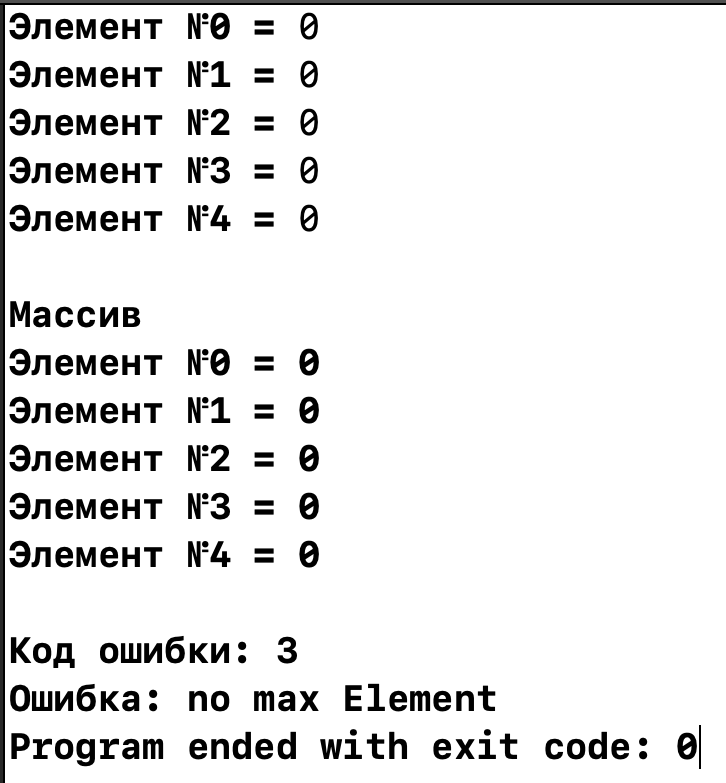
}

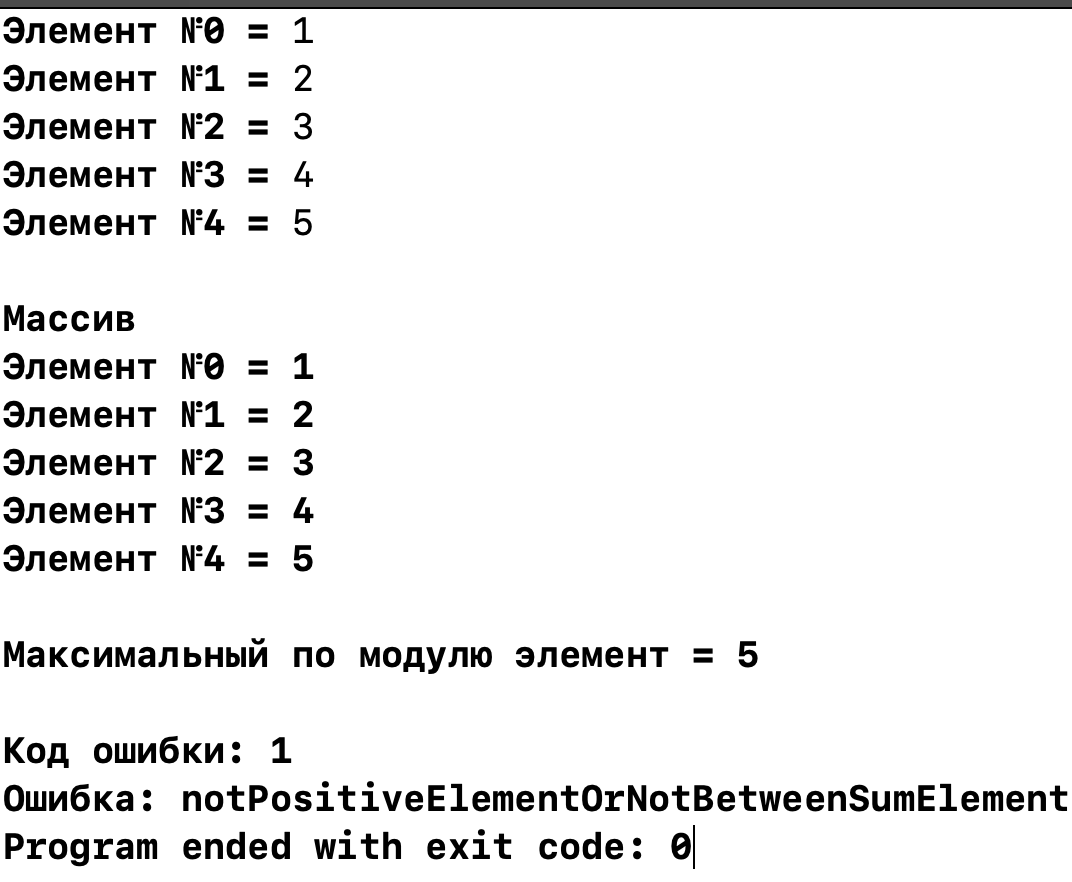
return 0;

}

Скриншот с результатами:







1. **Вывод**

В ходе лабораторной работы я получил практические навыки использования исключений в языке С++