|  |  |
| --- | --- |
|  | **Министерство науки и высшего образования Российской Федерации**  **Федеральное государственное бюджетное образовательное учреждение**  **высшего образования**  **«Московский государственный технический университет**  **имени Н.Э. Баумана**  **(национальный исследовательский университет)»**  **(МГТУ им. Н.Э. Баумана)** |

##### ФАКУЛЬТЕТ ИУ «Информатика и системы управления»

**КАФЕДРА ИУ8 «Компьютерная безопасность*»***

**ЛАБОРАТОРНАЯ РАБОТА №6**

**Название лабораторной работы: Исключения.**

**по курсу: Алгоритмические языки.**

**Группа ИУ8 – 24**

**Студент: Полонский Сергей**

**Преподаватель: Барыкин Дмитрий**

**Дата: 01.05.2024**

**ОТЧЕТ:**

**1)** **Цель работы:** научиться работать с исключениями.

**2)** **УСЛОВИЕ ЗАДАЧИ:**

Реализовать класс «Простое число». Поля: значение числа. Конструкторы: конструктор для инициализации полей. При необходимости добавить свои функции. При недопустимых параметрах объекта (число непростое) создается исключение. Для исключения создать класс, производный от invalid\_argument. Дополнительно созданный класс для исключения кроме наследуемых полей (необходимо их также использовать) содержит поля класса «Простое число» для недопустимого объекта и метод для печати всех данных. Продемонстрировать работу с объектами в случае возникновения исключения и его обработки и без возникновения исключения.

**3) Текст программы с коментариями:**

#include <iostream>

using namespace std;

class exeption : public invalid\_argument { // класс исключение, потомок класса invalid\_argument

public:

int t; // число

exeption(const char\* er,int j): invalid\_argument(er) // конструктор класса

{

t = j;

}

void print() { // метод вывода числа

cout << t;

}

};

class pr\_ch { // класс простое число

public:

int a;

pr\_ch(); // нулевой конструктор

pr\_ch(int b); // конструктор с параметрами

void print(); // метод print

};

pr\_ch::pr\_ch() {

a = 0;

}

pr\_ch::pr\_ch(int b) {

int k = 0;

for (int i = 2; i<= floor(sqrt(b)); ++i) if (b % i == 0) ++k; // подсчет количества делителей число

if (k > 0) throw exeption("Ошибка",b); // если k > 0, то есть делителей больше двух, то создается класс исключения и ссылку на него выбрасывает throw, если throw срабатывает, программа в конструкторе дальше не идет

a = b;

}

void pr\_ch::print() {

cout << a;

}

int main()

{

setlocale(LC\_ALL, "Rus"); // вывод в консоль на русском

try{ // блок контроля

pr\_ch obj(14);

obj.print();

}

catch (exeption& e) { // блок обработки исключения

e.print();

cout << endl;

cout << e.what(); // what единственный метод класса invalid\_argument, он служит для вывода сообщения об ошибке

}

}

**Вывод:** я научился работать с исключениями.