**Лабораторная работа № 3**

**Тема:** «Обработка исключительных ситуаций на языке С++»

**Цель:** «Изучить приемы обработки исключительных ситуаций при помощи классов на языке С++»

**Задание:**

1. Дополнить класс **Class\_MLib**, из предыдущей работы методами для вычислений значений тангенса и котангенса. Аргументы этих методов должны задаваться в градусах. Точность вычислений значений тангенса и котангенса: четыре знака после запятой. Если значение тангенса или котангенса может быть представлено как целое число, не показывать нули после запятой. В данных методах класса **Class\_MLib** предусмотреть инициацию исключительной ситуации ( деление на ноль ) при помощи оператора **throw** и класса **Class\_MsgError**, обрабатывающего исключительные ситуации.

2. Класс **Class\_MsgError** необходимо разработать самостоятельно. Он должен содержать два метода: **GetMsgCode** ( возвращающий код ошибки ) и **ShowMsg** ( который выводит сообщение об ошибке на экран ).

3. Написать консольную программу, организующую ввод значения угла ( в градусах ) и вычисляющую значения синуса, косинуса, тангенса и котангенса этого угла с использованием экземпляра класса **Class\_Mlib**. Предусмотреть обработку исключительной ситуации ( деление на ноль ) с выводом кода ошибки и сообщением об ошибке.

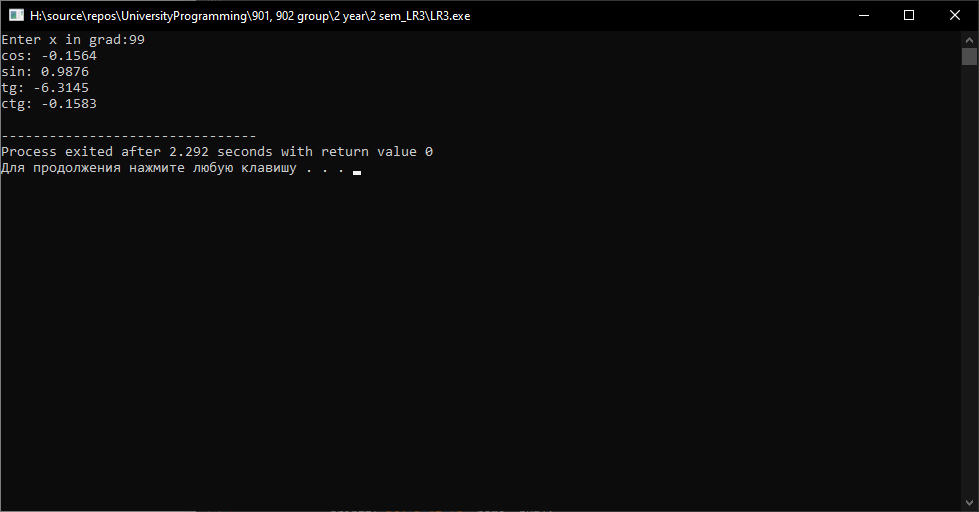


Рисунок 3.1 результат работы программы

Рисунок 3.2 результат работы программы

Код:

#define \_USE\_MATH\_DEFINES

#include <stdio.h>

#include <iostream>

#include <math.h>

#include <string>

using namespace std;

class Class\_MsgError{

public:

Class\_MsgError(const char\* msg="Unknown Exception", int code=-1){

\_code=code;

\_msg=msg;

}

int GetMsgCode() {

return \_code;

}

char\* ShowMsg() {

printf("Caughted error, code: %d, error: %s\n",GetMsgCode(),\_msg);

}

private:

const char\* \_msg;

int \_code;

};

class Class\_MLib {

public:

static Class\_MLib\* getInstance() {

if (instance)

instance = new Class\_MLib();

return instance;

}

double get\_cos(double x) {

return (int)(cos((x \* M\_PI) / 180) \* 10000) / 10000.;

}

double get\_sin(double x) {

return (int)(sin((x \* M\_PI) / 180) \* 10000) / 10000.;

}

double get\_tg(double x) {

double cos = get\_cos(x);

double sin = get\_sin(x);

if (cos == 0)

throw Class\_MsgError("division by 0", 1);

return (int)(sin / cos \* 10000) / 10000.;

}

double get\_ctg(double x) {

double cos = get\_cos(x);

double sin = get\_sin(x);

if (sin == 0)

throw Class\_MsgError("division by 0", 1);

return (int)(cos / sin \* 10000) / 10000.;

}

static void output(double num,const char\* name) {

if (modf(num, nullptr) == 0)

printf("%s: %.0f \n",name, num);

else

printf("%s: %.4f \n",name, num);

}

private:

Class\_MLib() {

}

~Class\_MLib() {

}

static Class\_MLib\* instance;

};

Class\_MLib\* Class\_MLib::instance;

int main() {

Class\_MLib\* math = Class\_MLib::getInstance();

double x;

cout << "Enter x in grad:";

cin >> x;

double cos = 0.0;

double sin = 0.0;

double tg = 0.0;

double ctg = 0.0;

cos = math->get\_cos(x);

sin = math->get\_sin(x);

math->output(cos, "cos");

math->output(sin, "sin");

try{

tg = math->get\_tg(x);

math->output(tg, "tg");

}

catch(Class\_MsgError &error){

error.ShowMsg();

}

try{

ctg = math->get\_ctg(x);

math->output(ctg, "ctg");

}

catch(Class\_MsgError &error){

error.ShowMsg();

}

return 0;

};

Вывод: в ходе работы мы изучили приемы обработки исключительных ситуаций при помощи классов на языке С++»