LabProgramming\_week\_3\_4\_5

7352396 – Ibrahim Siddiqui

LabProgramming\_week\_3\_4\_5.cpp:

#include "Point.h"

#include <iostream>

#include <string>

#include <fstream>

void Lab03\_01();

void Lab03\_02();

void Lab03\_06();

void Lab03\_07();

void Lab03\_10();

void Lab04\_05();

void Lab04\_07();

void Lab04\_08();

void Lab04\_11();

void Lab04\_12();

void Lab05\_01();

void Lab05\_07();

void Lab05\_08();

void Lab05\_09();

void Lab05\_10();

void second(int x) throw (int);

void first(int x);

int main()

{

std::cout << "Lab 3 starts" << std::endl;

std::cout << std::endl;

std::cout << "Lab03-01" << std::endl;

std::cout << std::endl;

Lab03\_01();

system("Pause");

std::cout << std::endl;

std::cout << "Lab03-02" << std::endl;

std::cout << std::endl;

Lab03\_02();

system("Pause");

std::cout << std::endl;

std::cout << "Lab03-06" << std::endl;

std::cout << std::endl;

Lab03\_06();

system("Pause");

std::cout << std::endl;

std::cout << "Lab03-07" << std::endl;

std::cout << std::endl;

Lab03\_07();

system("Pause");

std::cout << std::endl;

std::cout << "Lab03-10" << std::endl;

std::cout << std::endl;

Lab03\_10();

system("Pause");

std::cout << std::endl;

std::cout << "Lab 4 starts" << std::endl;

std::cout << std::endl;

std::cout << "Lab04-05" << std::endl;

std::cout << std::endl;

Lab04\_05();

system("Pause");

std::cout << std::endl;

std::cout << "Lab04-07" << std::endl;

std::cout << std::endl;

Lab04\_07();

system("Pause");

std::cout << std::endl;

std::cout << "Lab04-08" << std::endl;

std::cout << std::endl;

Lab04\_08();

system("Pause");

std::cout << std::endl;

std::cout << "Lab04-11" << std::endl;

std::cout << std::endl;

Lab04\_11();

system("Pause");

std::cout << std::endl;

std::cout << "Lab04-12" << std::endl;

std::cout << std::endl;

Lab04\_12();

system("Pause");

std::cout << std::endl;

std::cout << "Lab 5 starts" << std::endl;

std::cout << std::endl;

std::cout << "Lab05-01" << std::endl;

std::cout << std::endl;

Lab05\_01();

system("Pause");

std::cout << std::endl;

std::cout << "Lab05-07" << std::endl;

std::cout << std::endl;

Lab05\_07();

system("Pause");

std::cout << std::endl;

std::cout << "Lab05-08" << std::endl;

std::cout << std::endl;

Lab05\_08();

system("Pause");

std::cout << std::endl;

std::cout << "Lab05-09" << std::endl;

std::cout << std::endl;

Lab05\_09();

system("Pause");

std::cout << std::endl;

std::cout << "Lab05-10" << std::endl;

std::cout << std::endl;

Lab05\_10();

system("Pause");

std::cout << std::endl;

return 0;

}

void Lab03\_01()

{

}

void Lab03\_02()

{

}

void Lab03\_06()

{

/\*

\* Rectangle::Rectangle(int len, int wid)

\* :length(len), height(wid)

\* {

\*

\* }

\*/

}

void Lab03\_07()

{

/\*

\* There is no default constructor so Sample second(); wont work.

\* There is no parameterized constructor so Sample first(4); also wont work

\* Finally Sample third; needs brackets

\*/

}

void Lab03\_10()

{

Point p1(2.0,4.0);

p1.print();

p1.PointPos(3.4,2.4);

p1.distance(3.4, 2.4);

}

void Lab04\_05()

{

std::string st1 = "";

std::string st2 = "";

std::string stFinal = "";

std::cout << "Please enter a string" << std::endl;

std::getline(std::cin, st1);

std::cout << std::endl;

std::cout << "Please enter another string" << std::endl;

std::getline(std::cin, st2);

std::cout << std::endl;

for (int i = 0; i < st1.length(); i++)

{

for (int j = 0; j < st2.length(); j++)

{

if(st1[i] == st2[j])

{

stFinal = stFinal + st1[i];

break;

}

}

}

for (int i = 0; i < stFinal.length()-1; i++)

{

for (int j = 0; j < stFinal.length()-1; j++)

{

if (i==j)

{

stFinal.erase(j, 1);

}

}

}

std::cout << stFinal << std::endl;

}

void Lab04\_07()

{

int num = 0;

int input = 0;

std::ifstream file;

file.open("lab4\_07.txt");

if (!file.is\_open())

{

std::cout << "File could not be opened" << std::endl;

return;

}

std::cout << "Please enter the number you are looking for" << std::endl;

std::cin >> input;

while (file >> num)

{

if (input == num)

{

std::cout << "The number exists" << std::endl;

}

else

{

std::cout << "The number does not exist" << std::endl;

break;

}

}

file.close();

}

void Lab04\_08()

{

char character;

std::string letters = "";

std::ifstream file;

file.open("lab04\_08.txt");

std::getline(file, letters);

std::cout << "Please enter a Letter that you want to Delete" << std::endl;

std::cin >> character;

if (character >= 97)

{

character -= 32;

}

if (!file.is\_open())

{

std::cout << "File could not be opened" << std::endl;

return;

}

for (int i = 0; i < letters.length(); i++)

{

if (character == letters[i] )

{

for (int j = i; j < letters.size(); j++)

{

letters[j] = letters[j + 1];

}

}

}

letters[letters.length()] = char(0);

letters[letters.length()-1] = char(0);

std::cout << letters;

}

void Lab04\_11()

{

}

void Lab04\_12()

{

std::string s1 = "";

std::string s2 = "";

std::string sub1 = "";

std::string sub2 = "";

std::cout << "Please enter a string" << std::endl;

std::getline(std::cin, s1);

std::cout << std::endl;

std::cout << "Please enter another string" << std::endl;

std::getline(std::cin, s2);

std::cout << std::endl;

if (s2.length() % 2 == 0)

{

sub1 = s2.substr(0, s2.length() / 2);

sub2 = s2.substr(s2.length() / 2, s2.length());

s2 = sub1 + s1 + sub2;

std::cout << s2 << std::endl;

}

else if (s2.length() % 2 > 0)

{

s2 = s2 + s2[s2.length()-1];

sub1 = s2.substr(0, s2.length() / 2);

sub2 = s2.substr(s2.length() / 2, s2.length());

s2 = sub1 + s1 + sub2;

std::cout << s2 << std::endl;

}

}

void Lab05\_01()

{

// check the word document

Diagram, schematic

Description automatically generated

}

void Lab05\_07()

{

/\*

\* Sample& operator+=(int value)

\* {

\* add(value);

\* return \*this;

\* }

\*

\*/

}

void Lab05\_08()

{

}

void Lab05\_09()

{

//int value = 30;

//if (value > 20) throw value;

//std::cout << value;

/\*return 0;\*/

/\*

\* An error is thrown because the value is greater than 20.

\* the value is thrown but abort is called before the cout can be run

\*/

}

void Lab05\_10()

{

try

{

first(10);

}

catch (int value)

{

std::cout << value << std::endl;

}

/\*

\* So first the try for first(10); is run. the first(10); function is called, in that function

\* another try catch is run. in the try the second(1200); function is called. in that function the

\* if statement is run. the if statement throws x because x is greater than 1000. Then we exit out of second(1200);

\* and go onto the catch of first(10); because second throws a value. in the catch we throw another value x\*10.

\* Now we finally go to the original catch which is run because the catch in the first function throws

\* an int which is the condition for the original catch. the catch then outputs the value that is thrown.

\*

\*/

}

void second(int x) throw (int)

{

if (x > 1000) throw x;

}

void first(int x)

{

try

{

second(1200);

}

catch (...)

{

throw x \* 10;

}

}

Point.cpp:

#include "Point.h"

Point::Point()

:x(1.0),y(1.0)

{

}

Point::Point(double x, double y)

:x(x), y(y)

{

}

void Point::print()

{

std::cout << "The coordinates of the point are " << "(" << x << "," << y << ")" << std::endl;

}

void Point::PointPos(double x2, double y2)

{

double x1 = x;

double y1 = y;

if (x1 < x2 && y1 < y2)

{

std::cout << "Point" << "(" << x1 << "," << y1 << ")" << "is to the Bottom right of Point " << "(" << x2 << "," << y2 << ")" << std::endl;

}

else if (x1 < x2 && y1 > y2)

{

std::cout << "Point" << "(" << x1 << "," << y1 << ")" << "is to the Top Right of Point " << "(" << x2 << "," << y2 << ")" << std::endl;

}

else if (x1 > x2 && y1 < y2)

{

std::cout << "Point" << "(" << x1 << "," << y1 << ")" << "is to the Bottom left of Point " << "(" << x2 << "," << y2 << ")" << std::endl;

}

else if (x1 > x2 && y1 > y2)

{

std::cout << "Point" << "(" << x1 << "," << y1 << ")" << "is to the Top left of Point " << "(" << x2 << "," << y2 << ")" << std::endl;

}

else if (x1==x2 && y1 > y2)

{

std::cout << "Point" << "(" << x1 << "," << y1 << ")" << "is Above Point " << "(" << x2 << "," << y2 << ")" << std::endl;

}

else if (x1==x2 && y1 < y2)

{

std::cout << "Point" << "(" << x1 << "," << y1 << ")" << "is Below Point " << "(" << x2 << "," << y2 << ")" << std::endl;

}

else if (x1 > x2 && y1==y2)

{

std::cout << "Point" << "(" << x1 << "," << y1 << ")" << "is to the Right of Point " << "(" << x2 << "," << y2 << ")" << std::endl;

}

else if (x1 < x2 && y1 == y2)

{

std::cout << "Point" << "(" << x1 << "," << y1 << ")" << "is to the Left of Point " << "(" << x2 << "," << y2 << ")" << std::endl;

}

}

void Point::distance(double x2, double y2)

{

double distance = 0.0;

double x1 = x;

double y1 = y;

distance = sqrt((x2-x1) \* 2 + (y2-y1) \* 2);

std::cout << "The distance between point 1 and point 2 is " << distance << std::endl;

}

Point.h:

#pragma once

#include <iostream>

class Point

{

private:

double x;

double y;

public:

Point();

Point(double x, double y);

void print();

void PointPos(double x2, double y2);

void distance(double x2, double y2);

};