# Lab 11

Problem: Create a program to draw certain shapes with user input to test polymorphism and inheritance.

## Source Files:

main:

#include "circle.h"

#include "rectangle.h"

int main(){

shape\* shapes[5];

shapes[0]=new circle(20,100,50);

shapes[1]=new rectangle(10,10,50,50);

shapes[2]=new rectangle(5,5,74,25);

shapes[3]=new rectangle(7,2,22,43);

shapes[4]=new circle(30,148,102);

for(int i=0;i<5;i++)

shapes[i]->draw();

cout<<endl<<endl<<endl;

shapes[0]->moveTo(148,102);

shapes[1]->moveTo(24,72);

shapes[2]->moveTo(60,20);

shapes[3]->erase();

shapes[4]->erase();

return 0;

}

rectangle.h:

#pragma once

#include "shape.h"

class rectangle :

public shape

{

public:

rectangle(void);

~rectangle(void);

protected:

int width;

int height;

public:

rectangle(int width, int height, int x0, int y0);

virtual void draw(void);

};

rectangle.cpp:

#include "rectangle.h"

rectangle::rectangle(void)

: width(0)

, height(0)

{

}

rectangle::~rectangle(void)

{

}

rectangle::rectangle(int width, int height, int x0, int y0)

:width(width),height(height),shape(x0,y0)

{

}

void rectangle::draw(void)

{

cout<<"Drawing a rectangle at midpoint ("<<x0<<","<<y0<<") with width "<<width<<" and height "<<height<<"\n";

}

circle.h:

#pragma once

#include "shape.h"

class circle :

public shape

{

public:

circle(void);

~circle(void);

protected:

int radius;

public:

circle(int radius, int x0, int y0);

virtual void draw(void);

};

circle.cpp:

#include "circle.h"

circle::circle(void)

: radius(0)

{

}

circle::~circle(void)

{

}

circle::circle(int radius, int x0, int y0)

:radius(radius),shape(x0,y0)

{

}

void circle::draw(void)

{

cout<<"Drawing a circle with radius "<<radius<<" at midpoint ("<<x0<<","<<y0<<")\n";

}

shape.h:

#pragma once

#include <iostream>

using namespace std;

class shape

{

public:

shape(void);

~shape(void);

protected:

int x0;

int y0;

public:

shape(int x0, int y0);

virtual void draw(void) = 0; //virtual means it can be overriden, 0 means its pure(prototype)

void erase(void);

void moveTo(int x, int y);

};

shape.cpp:

#include "shape.h"

shape::shape(void)

: x0(0)

, y0(0)

{

}

shape::~shape(void)

{

}

shape::shape(int x0, int y0)

:x0(x0),y0(y0)

{

}

void shape::erase(void)

{

cout<<"Changing pen to background color\n";

draw();

cout<<"Changing pen to foreground color\n";

}

void shape::moveTo(int x, int y)

{

erase();

x0=x;

y0=y;

draw();

}

## Sample Run:

Drawing a circle with radius 20 at midpoint (100,50)

Drawing a rectangle at midpoint (50,50) with width 10 and height 10

Drawing a rectangle at midpoint (74,25) with width 5 and height 5

Drawing a rectangle at midpoint (22,43) with width 7 and height 2

Drawing a circle with radius 30 at midpoint (148,102)

Changing pen to background color

Drawing a circle with radius 20 at midpoint (100,50)

Changing pen to foreground color

Drawing a circle with radius 20 at midpoint (148,102)

Changing pen to background color

Drawing a rectangle at midpoint (50,50) with width 10 and height 10

Changing pen to foreground color

Drawing a rectangle at midpoint (24,72) with width 10 and height 10

Changing pen to background color

Drawing a rectangle at midpoint (74,25) with width 5 and height 5

Changing pen to foreground color

Drawing a rectangle at midpoint (60,20) with width 5 and height 5

Changing pen to background color

Drawing a rectangle at midpoint (22,43) with width 7 and height 2

Changing pen to foreground color

Changing pen to background color

Drawing a circle with radius 30 at midpoint (148,102)

Changing pen to foreground color

Press any key to continue . . .