Описати клас «Figure», який є базовим для класів «Rectangle» і «Circle» і містить : кількість сторін і масив довжин сторін та віртуальні методи для знаходження площі та периметру.

#pragma once

#include <string>

using namespace std;

class Figure

{

private:

int sidesCount;

double\* sides;

void set\_sidesCount(int value);

bool isIndexCorrect(int index);

public:

int get\_sidesCout();

double get\_side(int index);

void set\_side(int index, double value);

Figure(int sidesCout, double firstSide1,...);

Figure(int sidesCout);

void input();

string toString();

virtual double S() = 0; //Чисто віртуальні (абстрактні ) методи

virtual double P() = 0;

~Figure();

};

//==========================================

#include "stdafx.h"

#include "Figure.h"

#include <iostream>

using namespace std;

Figure::~Figure()

{

delete[] sides;

}

void Figure::set\_sidesCount(int value)

{

if (value > 0)

{

sidesCount = value;

sides = new double[value];

}

else

{

throw "error";

}

}

bool Figure::isIndexCorrect(int index)

{

return index>=0 && index<sidesCount;

}

int Figure::get\_sidesCout()

{

return sidesCount;

}

double Figure::get\_side(int index)

{

if (isIndexCorrect(index))

{

return sides[index];

}

else

throw "index is incorrect";

}

void Figure::set\_side(int index, double value)

{

if (isIndexCorrect(index))

{

if (value>=0)

{

sides[index] = value;

}

else

{

throw "value is negative";

}

}

else

{

throw "index is incorrect";

}

}

Figure::Figure(int sidesCout, double firstSide1, ...)

{

set\_sidesCount(sidesCount);

double\* data = &firstSide1;

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

{

sides[i] = data[i];

}

}

Figure::Figure(int sidesCout)

{

set\_sidesCount(sidesCount);

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

{

sides[i] = 0;

}

}

void Figure::input()

{

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

{

printf("Side %d=", i);

double temp;

cin >> temp;

set\_side(i,temp);

}

}

string Figure::toString()

{

string temp = to\_string(sides[0]);

for (int i = 1; i < sidesCount; i++)

{

temp+=", "+ to\_string(sides[i]);

}

return temp;

}

//=============================

#pragma once

#include "Figure.h"

class Rectangle :

public Figure

{

public:

int get\_a(); //side[0]

void set\_a(double value);

int get\_b(); //side[1]

void set\_b(double value);

Rectangle(double a, double b);

Rectangle(double a);

Rectangle();

double S(); //Перевизначення чисто віртуальних методів

double P();

~Rectangle();

};

//======================================

#include "stdafx.h"

#include "Rectangle.h"

int Rectangle::get\_a()

{

return get\_side(0); //у side[0] зберігаємо а

}

void Rectangle::set\_a(double value)

{

set\_side(0, value);

}

int Rectangle::get\_b()

{

return get\_side(1); //у side[1] зберігаємо b

}

void Rectangle::set\_b(double value)

{

set\_side(1,value);

}

Rectangle::Rectangle(double a, double b):Figure(2,a,b)

{

}

Rectangle::Rectangle(double a):Rectangle(a,a)

{

}

Rectangle::Rectangle():Figure(2)

{

}

double Rectangle::S()

{

return get\_a()\*get\_b();

}

double Rectangle::P()

{

return 2\*(get\_a()+get\_b());

}

Rectangle::~Rectangle()

{

}

//=======================================

#pragma once

#include "Figure.h"

class Circle :

public Figure

{

public:

double get\_radius();

void set\_radius(double value);

Circle(double radius);

Circle();

double S(); //Перевизначення чисто віртуальних методів

double P();

~Circle();

};

//=========================================

#include "stdafx.h"

#include "Circle.h"

double Circle::get\_radius()

{

return get\_side(0); //side[0]

}

void Circle::set\_radius(double value)

{

set\_side(0,value);

}

Circle::Circle(double radius):Figure(1,radius)

{

}

Circle::Circle():Circle(0)

{

}

double Circle::S()

{

return 3.14\*pow(get\_radius(),2);

}

double Circle::P()

{

return 2\*3.14\*get\_radius();

}

Circle::~Circle()

{

}

//====================== main ==================

// ConsoleApplication42.cpp : Defines the entry point for the console application.

//

#include "stdafx.h"

#include <iostream>

#include "Rectangle.h"

#include "Circle.h"

#include "Figure.h"

using namespace std;

int main()

{

Figure\*\* figures=new Figure\*[5];

figures[0] = new Rectangle(2,8); //У покажчик предка (Figure\*) запусую адресу новоствореного нащадка (Rectangle)

figures[1] = new Circle(6);

figures[2] = new Circle(12);

figures[3] = new Rectangle(5, 2);

figures[4] = new Circle(9);

double totalS = 0;

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

{

totalS += figures[i]->S();

}

//-------------------------

printf("s=%f", totalS);

system("pause");

return 0;

}