**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ**

### **Федеральное государственное автономное образовательное учреждение высшего образования**

**«Санкт-Петербургский государственный университет  
аэрокосмического приборостроения»**

КАФЕДРА № 43

ОТЧЕТ   
ЗАЩИЩЕН С ОЦЕНКОЙ

ПРЕПОДАВАТЕЛЬ

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Старший преподаватель |  |  |  |  |
| должность, уч. степень, звание |  | подпись, дата |  | инициалы, фамилия |

|  |
| --- |
| ОТЧЕТ О ЛАБОРАТОРНОЙ РАБОТЕ |
| « НАСЛЕДОВАНИЕ КЛАССОВ, БАЗОВЫЙ КЛАСС, ПРОИЗВОДНЫЙ КЛАСС » |
| по дисциплине: ОБЪЕКТНО-ОРИЕНТИРОВАННОЕ ПРОГРАММИРОВАНИЕ |
|  |

РАБОТУ ВЫПОЛНИЛ

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| СТУДЕНТ ГР. № | 4131 |  |  |  | В.А. Алексеев |
|  |  |  | подпись, дата |  | инициалы, фамилия |

1. Задание

В работе необходимо реализовать базовый класс графический объект (название фигуры, координаты, радиус или длина), на основе него создать производные классы Круг, Квадрат. В базовом предусмотреть конструктор для установки начальных значений полей и метод вывода параметров фигуры. В классах Круг и Квадрат предусмотреть методы для расчета площади фигуры, периметра (длины окружности). Создать объекты производных классов. Продемонстрировать работу всех методов, реализуемых в классах.

1. Листинг

Header.h

#pragma once

class object {

protected:

char name[10];

int x;

int y;

int a;

public:

object(char\*, int, int, int);

object();

void show();

};

class square : public object {

public:

square(char\*, int, int, int);

void showarea();

void showperimeter();

};

class circle : public object {

public:

circle(char\*, int, int, int);

void showarea();

void showperimeter();

};

Source.cpp

#include "Header.h"

#include<iostream>

#include <windows.h>

#define \_USE\_MATH\_DEFINES

#include <math.h>

using namespace std;

object::object(char\* ar, int x, int y, int l) {

strcpy\_s(name, ar);

this->x = x;

this->y = y;

this->a = l;

}

object::object() {

strcpy\_s(name, "dds");

this->x = 6;

this->y = 7;

this->a = 6;

}

void object::show() {

cout << "Имя : " << name << endl;

cout << "Кордината центра фигуры по Х : " << this->x << endl;

cout << "Кордината центра фигуры по Y : " << this->y << endl;

cout << "Длина фигуры или радиус : " << this->a << endl;

}

void square::showarea() {

cout << "Площадь квадрата равна : " << a \* a << endl;

}

void square::showperimeter() {

cout << "Периметр квадрата равен : " << a \* 4 << endl;

}

square::square(char\* a, int x, int y, int l) :

object(a, x, y, l) {

}

void circle::showarea() {

cout << "Площадь круга равна : " << M\_PI \* a \* a << endl;

}

void circle::showperimeter() {

cout << "Периметр круга равен : " << M\_PI \* a \* 2 << endl;

}

circle::circle(char\* a, int x, int y, int l) :

object(a, x, y, l) {

}

int main() {

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

char name[10];

int x, y, l;

cout << "Создается квадрат.Введите название фигуры : ";

cin >> name;

cout << "Введите абсциссу центра квадрата : ";

cin >> x;

cout << "Введите ординату центра квадрата : ";

cin >> y;

cout << "Введите длину квадрата : ";

cin >> l;

square c(name, x, y, l);

cout << "Создается круг.Введите название фигуры : ";

cin >> name;

cout << "Введите абсциссу центра кргуа : ";

cin >> x;

cout << "Введите ординату центра кргуа : ";

cin >> y;

cout << "Введите радиус круга : ";

cin >> l;

circle k(name, x, y, l);

c.show();

c.showarea();

c.showperimeter();

k.show();

k.showarea();

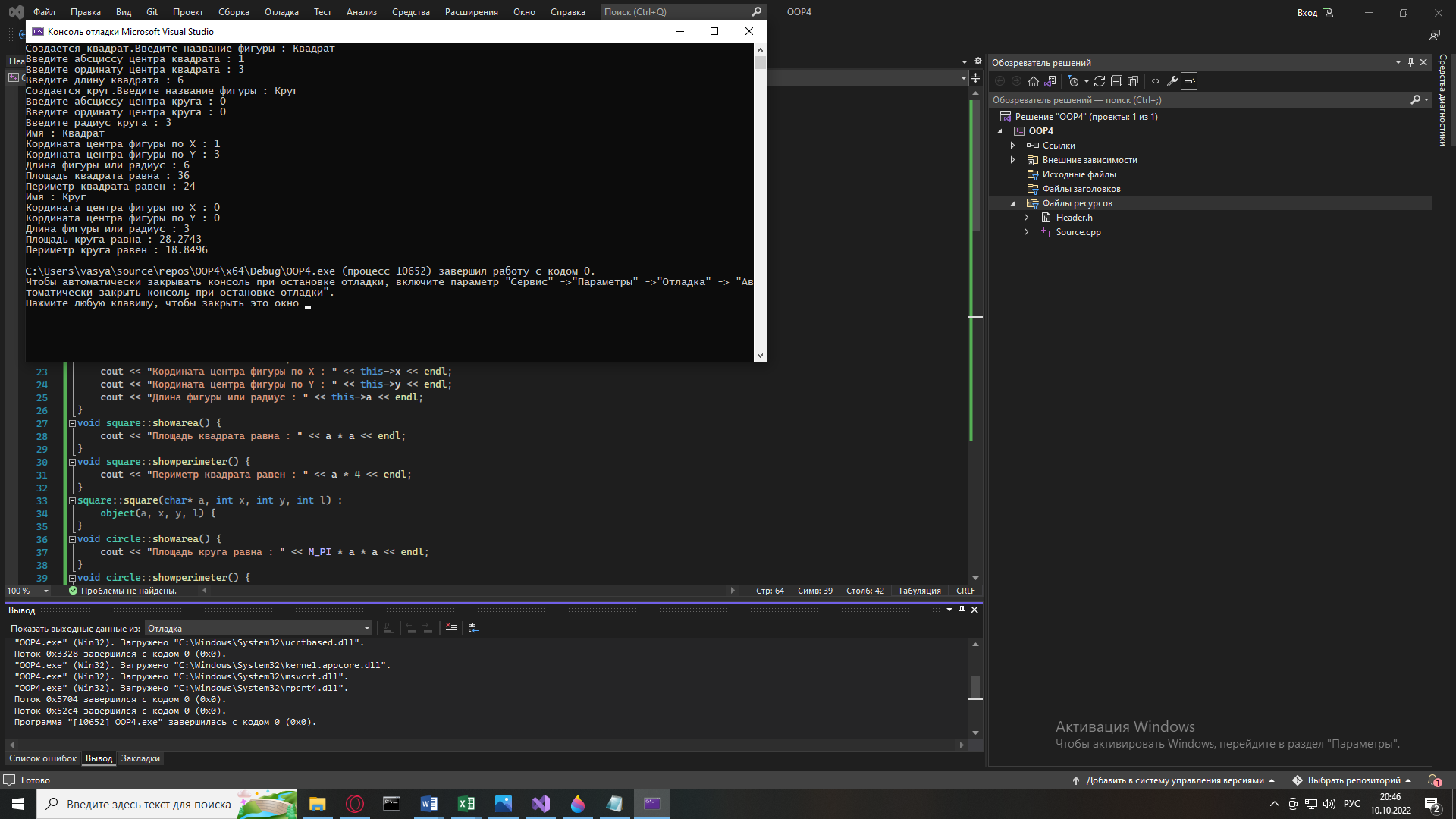
k.showperimeter();

return 0;

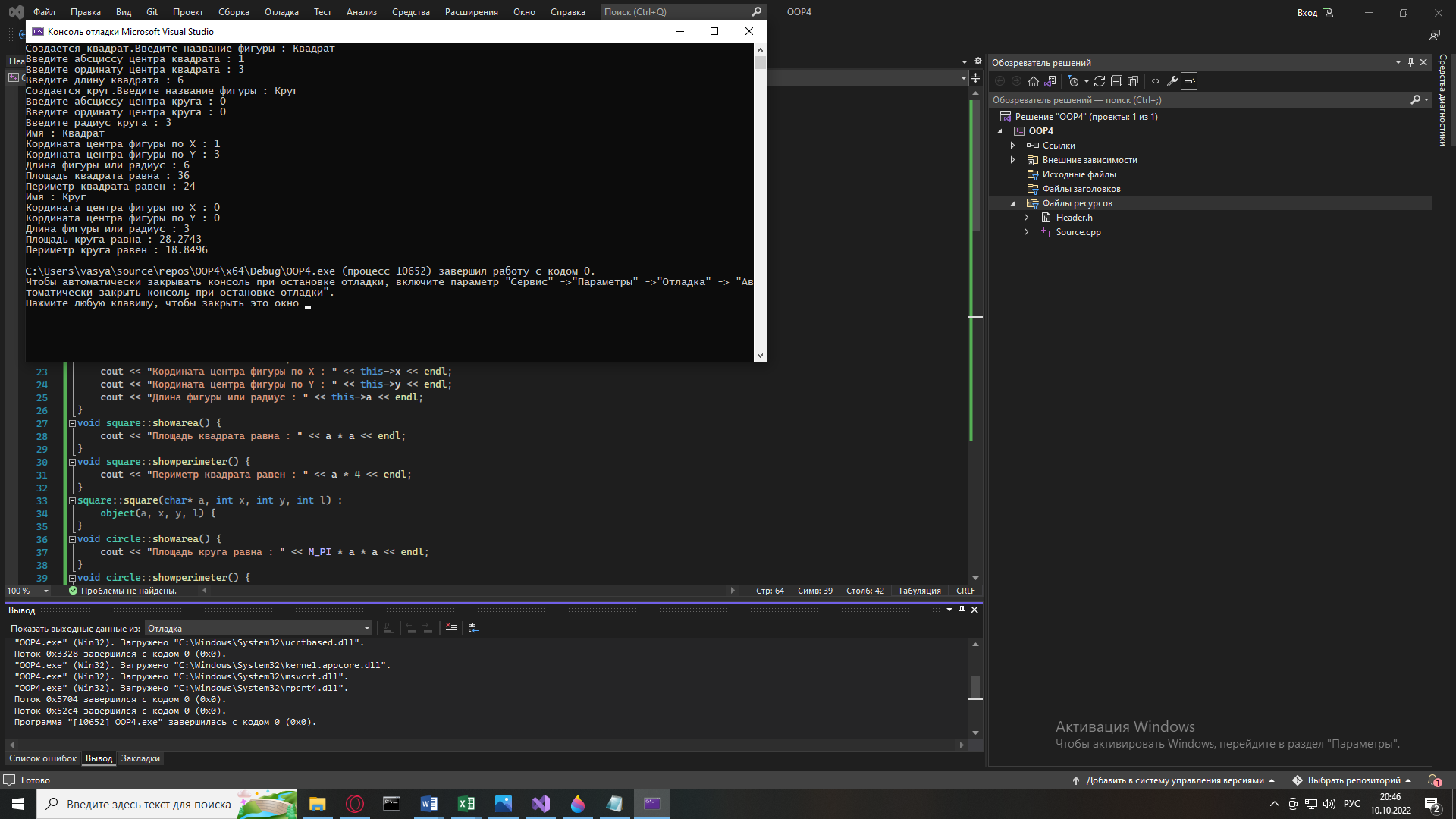
}

1. Результат работы

Входные данные



Вывод



1. Вывод

Я изучил и применил на практике механизм создания нового класса на основе уже существующего, а также варианты доступа к элементам базового класса из производного, закрепил знания по теме: Классы, наследование классов, варианты доступа.