



Министерство науки и высшего образования Российской Федерации  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Московский государственный технический университет  
имени Н.Э. Баумана  
(национальный исследовательский университет)»  
(МГТУ им. Н.Э. Баумана)

---

ФАКУЛЬТЕТ ИНФОРМАТИКА И СИСТЕМЫ УПРАВЛЕНИЯ

КАФЕДРА КОМПЬЮТЕРНЫЕ СИСТЕМЫ И СЕТИ (ИУ6)

НАПРАВЛЕНИЕ ПОДГОТОВКИ 09.03.03 Прикладная информатика

**О Т Ч Е Т**

по лабораторной работе № 8

Название: Наследование

Дисциплина: Объектно-ориентированное программирование

Студент

ИУ6-25 Б  
(Группа)

Душина И.А.  
(Подпись, дата) (И.О. Фамилия)

Преподаватель

Васильева С.А.  
(Подпись, дата) (И.О. Фамилия)

Москва, 2023

## Вариант 8

**Задание:** разработать и реализовать иерархию классов для описанных объектов предметной области, используя механизмы наследования.

Протестировать все методы каждого класса. Все поля классов должны быть скрытыми (private) или защищенными (protected). Методы не должны содержать операций ввода/вывода, за исключением процедуры, единственной задачей которой является вывод информации об объекте на экран.

Объект – круг. Поле: диаметр. Методы: процедура инициализации, процедура вывода информации об объекте на экран и функция вычисления площади круга.

Объект – сковорода. Поля: диаметр сковороды, материал покрытия. Методы: процедура инициализации, процедура вывода информации об объекте на экран и функция, определяющая, сколько котлет заданной площади одновременно поместится на сковороде.

В отчете привести диаграмму разработанных классов и объектную декомпозицию.

### Код программы:

```
#include <iostream>
#include <cmath>
#include <math.h>
using namespace std;
class circle {
protected:
    int diam;
public:
    circle(int ad) : diam(ad) {}
    circle() {}
    float square() {
        float sq; int r;
        r = diam / 2;
        sq = 3.14*pow(r, 2);
        return sq;
    }
    void print() {
        cout << "The diametr: " << diam<<endl;
        cout << "The square of circle: " << square();
    }
};
class pan :public circle {
private:
    char material[22];
public:
    pan(int ad, char am[22]):circle(ad){
        strcpy_s(material, am);
    }
    pan() {}
    void print() {
```

```

        circle::print();
        cout << "\nThe material: " << material<<endl;
    }
    int kotlet(float sq) {
        float Sq;
        Sq=circle::square();
        return trunc(Sq / sq);
    }
};
int main() {
    int d; float square; char m[22];
    cout << "Input the diametr: \n";
    cin >> d;
    cout << "The circle: \n----- \n";
    circle Krug(d);
    Krug.print();
    //-----
    cout << "\nInput the material: \n";
    cin.ignore();
    gets_s(m, 22);
    cout << "Input the square of meatballs: \n";
    cin >> square;
    pan Skov(d, m);
    cout << "The pan: \n----- \n";
    Skov.print();
    cout << "The amount of meatballs: " << Skov.kotlet(square);
    return 0;
}

```

```

Input the diametr:
14
The circle:
-----
The diametr: 14
The square of circle: 153.86
Input the material:
still
Input the square of meatballs:
3
The pan:
-----
The diametr: 14
The square of circle: 153.86
The material: still
The amount of meatballs: 51

```

Рисунок 1 – работающая версия программы

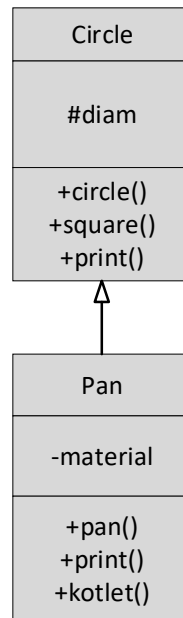


Рисунок 2 – блок-схема программы

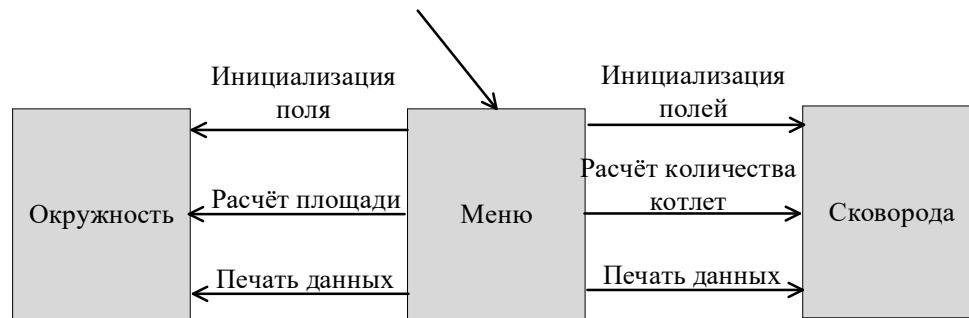


Рисунок 3 – объектная декомпозиция

**Вывод:** мы научились наследовать классы друг от друга и применять простой полиморфизм.