



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 **09.03.01 Информатика и вычислительная техника**

О Т Ч Е Т

по лабораторной работе № 8

Название: Наследование

Дисциплина: Объектно-ориентированное программирование

Студент

ИУ6-22Б

(Группа)

(Подпись, дата)

С.В. Астахов

(И.О. Фамилия)

Преподаватель

(Подпись, дата)

(И.О. Фамилия)

Москва, 2020

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 – хлеб. Поля: масса, стоимость. Методы: процедура инициализации, процедура вывода значений полей на экран, функция вычисления цены 100 грамм хлеба.

Объект – булочка с сыром. Поля: масса булочки, масса начинки, стоимость. Методы: процедура инициализации, процедура вывода значений полей на экран, функция определения массовой доли начинки в булочке и функция вычисления цены 100 грамм продукта.

В отчете привести диаграмму разработанных классов и объектную декомпозицию.

Исходный код

(файл Source.cpp)

```
#include <iostream>
using namespace std;
```

```
int main()
{
```

```
    class CBread {
    protected:
        int m;
        int price;
    public:
        void sett(int argM, int argPrice) {
            m = argM;
            price = argPrice;
        }
        void get() {
            printf("\n massa: %d price: %d", m, price);
        }
        double cost() {
            return (price * 100.0) / (m * 1.0);
        }
        CBread() {
            m = 0;
            price = 0;
        }
    };
    class CBun : public CBread {
```

```

protected:
    int ms;
public:
    float mPart() {
        return (ms * 1.0) / (m);
    }
    void sett(int argM, int argMs, int argPrice) {
        CBread::sett(argM, argPrice);
        ms = argMs;
    }
    CBun() :CBread() {
        ms = 0;
    }
};

int a1, a2, a3;
CBread bread;
CBun bun;

puts("Enter M and PRICE for bread");
scanf_s("%d %d", &a1, &a2);
bread.sett(a1, a2);
bread.get();
printf("\n cost for 100g: %3.2f", bread.cost());

puts("\n \n Enter M and M OF FILLING and PRICE for bun");
scanf_s("%d %d %d", &a1, &a2, &a3);
bun.sett(a1, a2, a3);
bun.get();
printf("\n cost for 100g: %3.2f", bun.cost());
printf("\n fillng mass part: %3.2f", bun.mPart());

return 0;
}

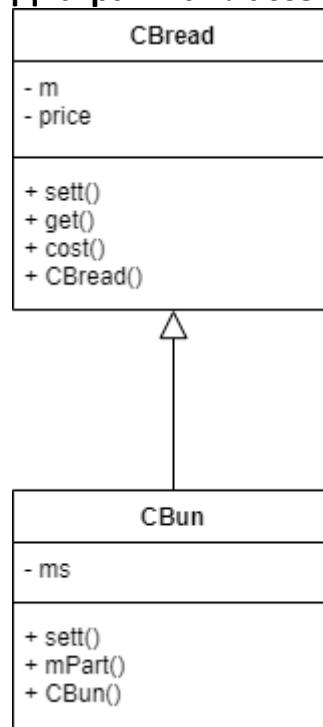
```

Тесты

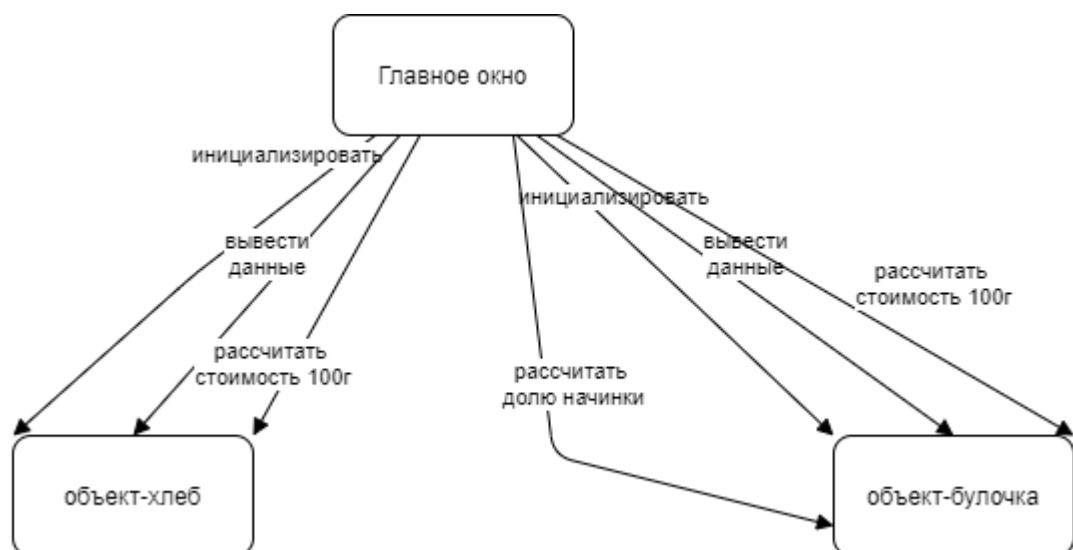
Входные данные	Ожидаемые выходные данные	Выходные данные
200 30 100 20 55	<p>Enter M and PRICE for bread 200 30</p> <p>massa: 200 price: 30 cost for 100g: 15.00</p> <p>Enter M and M OF FILLING and PRICE for bun 100 20 55</p> <p>massa: 100 price: 55 cost for 100g: 55.00 fillng mass part: 0.20</p>	<p>Enter M and PRICE for bread 200 30</p> <p>massa: 200 price: 30 cost for 100g: 15.00</p> <p>Enter M and M OF FILLING and PRICE for bun 100 20 55</p> <p>massa: 100 price: 55 cost for 100g: 55.00 fillng mass part: 0.20</p>
250 45 50 15 40	<p>Enter M and PRICE for bread 250 45</p> <p>massa: 250 price: 45 cost for 100g: 18.00</p> <p>Enter M and M OF FILLING and PRICE for bun 50 15 40</p> <p>massa: 50 price: 40 cost for 100g: 80.00 fillng mass part: 0.30</p>	<p>Enter M and PRICE for bread 250 45</p> <p>massa: 250 price: 45 cost for 100g: 18.00</p> <p>Enter M and M OF FILLING and PRICE for bun 50 15 40</p> <p>massa: 50 price: 40 cost for 100g: 80.00 fillng mass part: 0.30</p>
300 50 90 15 55	<p>Enter M and PRICE for bread 300 50</p> <p>massa: 300 price: 50 cost for 100g: 16.67</p> <p>Enter M and M OF FILLING and PRICE for bun 90 15 55</p>	<p>Enter M and PRICE for bread 300 50</p> <p>massa: 300 price: 50 cost for 100g: 16.67</p> <p>Enter M and M OF FILLING and PRICE for bun 90 15 55</p>

	massa: 90 price: 55 cost for 100g: 61.11 filling mass part: 0.17	massa: 90 price: 55 cost for 100g: 61.11 filling mass part: 0.17
--	--	--

Диаграмма классов



Объектная декомпозиция



Вывод

- В общем и целом объектная модель в С++ напоминает Delphi, однако стоит отметить, что в С++ существуют различные спецификаторы доступа при наследовании.