

Факультет Информационных систем и технологий  
Кафедра Безопасности информационных систем

**Дисциплина «Алгоритмизация и программирование»**

## **ОТЧЕТ**

**За практическое занятие №6**

**Тема: Проектирование классов с наследованием**

**Вариант: 7**

Выполнил  
Студент 2 курса, гр. ИБ-02з  
\_\_\_\_\_ Винокуров А.Ф.

Принял  
Доцент кафедры БИС  
\_\_\_\_\_ Медведев В.А.

## Тема: Проектирование классов с наследованием

**Цель:** получение практических навыков проектирования классов и исследование механизмов открытого и закрытого наследования в C++.

### Вариант 1.

А. Спроектировать структуру классов в соответствии с индивидуальным заданием (см. Приложение). Наследование осуществляется по типу **public**.

Б. Создать несколько объектов производных классов, задавая случайным образом их свойства.

В. Для определения *Вычисляемого показателя* использовать дружественную функцию.

№ п/п	Базовый класс	Производные классы	Вычисляемый показатель
7	Периферийное устройство компьютера	Клавиатура, мышь	Минимальная цена устройства

### Текст программы:

```
#include <iostream>

using namespace std;

class Peripheral {
    int id;
    double price;
public:
    Peripheral() { id = 0; price = 0; }
    Peripheral(int _id, double _price) { id = _id; price = _price; }
    double GetPrice() {
        return price;
    }
    int GetID() {
        return id;
    }
    ~Peripheral() {};

    friend double MinValuePrice(Peripheral[], int count);
};

class Mouse : public Peripheral {
    int dpi;
    int mouseKey;
public:
    Mouse(int _dpi, int _mousekey) : Peripheral(rand() % 100, rand() % 200 + 20) { dpi = _dpi; mouseKey = _mousekey; }
```

```

    }
    int GetDPI () {
        return dpi;
    }
    int GetMouseKey () {
        return mouseKey;
    }
};

class Keyboard : public Peripheral {
    int keyboardKey;
    bool mechanical;

public:
    Keyboard(int _keyboardKey, bool _mechanical) : Peripheral(rand() %
100, rand() % 200 + 20) { keyboardKey = _keyboardKey; mechanical
=_mechanical;
    }
    int GetKeyboardKey () {
        return keyboardKey;
    }
    bool GetMechanical () {

        return mechanical;
    }
};

int main ()
{ Mouse spaceX(30 + rand() % 100, 2 + rand() % 5); Mouse Virgin(30 + rand()
% 100, 2 + rand() % 7); Keyboard BladeZ(21 + rand() % 146, 0 + rand() % 2);
Keyboard Quality(13 + rand() % 146, 0 + rand() % 2); Peripheral arr[4] =
{spaceX, Virgin, BladeZ, Quality};

    cout << endl << "Mouse\t\t" << endl << endl << "Name\t\t" << "\tmouse DPI
" << "\tkey count \t" << "Price\t" << "ID\t" << endl;
    cout << "Space X: " << "\t\t" << spaceX.GetDPI () << "\t\t" <<
spaceX.GetMouseKey () << "\t\t" << spaceX.GetPrice () << "\t" << spaceX.GetID ()
<< endl;
    cout << "Virgin: " << "\t\t" << Virgin.GetDPI () << "\t\t" <<
Virgin.GetMouseKey () << "\t\t" << Virgin.GetPrice () << "\t" << Virgin.GetID ()
<< endl;
    cout << "\n\n" << "Keyboard\t\t" << "\n\n" << "Name\t\t" << "\tKey count "
<< "\tmechanical keys \t" << "Price\t" << "ID\t" << endl;
    cout << "BladeZ:" << "\t\t\t" << BladeZ.GetKeyboardKey () << "\t\t\t" <<
BladeZ.GetMechanical () << "\t\t" << BladeZ.GetPrice () << "\t" <<
BladeZ.GetID () << endl;
    cout << "Quality:" << "\t\t" << Quality.GetKeyboardKey () << "\t\t\t"
<< Quality.GetMechanical () << "\t\t" << Quality.GetPrice () << "\t" <<
Quality.GetID () << endl;

    cout << endl << "min value of price: " << MinValuePrice(arr, 4) << endl;
    return 0;
}

double MinValuePrice (Peripheral mass [], int count) {

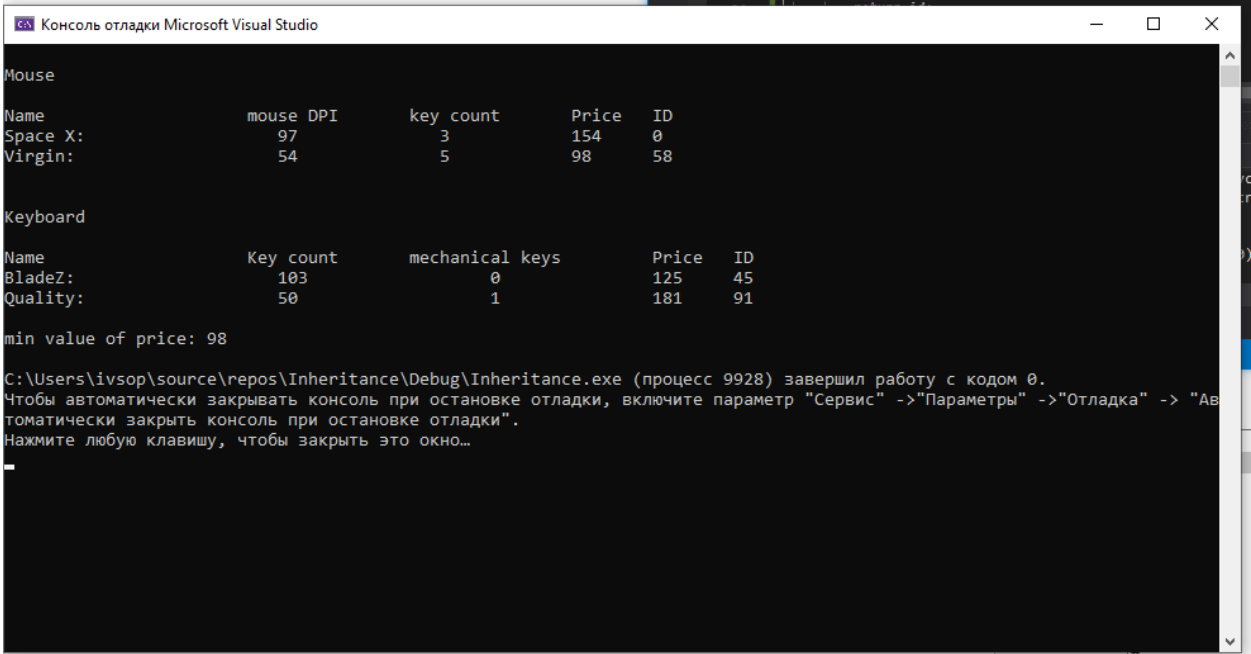
    double min = 0;
    min = mass[0].GetPrice ();
    for (int i = 1; i < count - 1; i++)
    {
        if (min > mass[i].GetPrice ())
        {
            min = mass[i].GetPrice ();
        }
    }
}

```

```
    }  
    }  
    return min;  
}
```

## Результат работы программы:

Результат работы программы:



```
Консоль отладки Microsoft Visual Studio  
Mouse  
Name          mouse DPI    key count    Price  ID  
Space X:      97          3            154    0  
Virgin:       54          5            98     58  
  
Keyboard  
Name          Key count    mechanical keys    Price  ID  
BladeZ:       103         0                 125    45  
Quality:      50          1                 181    91  
  
min value of price: 98  
  
C:\Users\ivso\source\repos\Inheritance\Debug\Inheritance.exe (процесс 9928) завершил работу с кодом 0.  
Чтобы автоматически закрывать консоль при остановке отладки, включите параметр "Сервис" ->"Параметры" ->"Отладка" -> "Автоматически закрывать консоль при остановке отладки".  
Нажмите любую клавишу, чтобы закрыть это окно...
```

Рисунок 1 Результат работы программы по нахождению минимальной цены при помощи дружественной функции

**Вывод:** в результате выполнения практической работы были получены навыки проектирования классов и исследованы механизмы открытого и закрытого наследования в C++.