

Сургай Р.В.
ИИУКУ-211

Контрольная работа №2

Вариант 15 (3)

Виртуальная функция — это функция, которая определяется в базовом классе, а любой производный (наследственный) класс может ее переопределить. Для объявления виртуальной функции используется ключевое слово *Virtual*.

Полиморфизм — это способность объектов использовать методы производного класса, который не существует на момент создания базового. Возможность объектов с одинаковыми спецификациями иметь разную реализацию.

Абстрактный класс — базовый класс, который не предполагает создание экземпляров. Абстрактные классы реализуют один из принципов ООП — полиморфизм.

Класс населенные пункты

```
#include <iostream>
```

```
#include <string>
```

```
using namespace std;
```

```
class Locality { // базовый класс
```

```
public:
```

```
    string name;
```

```
    int numResidents;
```

```
    Locality(string name, int nr) {
```

```
        this->name = name;
```

```
        this->numResidents = nr;
```

```
    }
```

```
};
```

```
class PublicTransport : public Locality {
```

```
private:
```

```
    int numOfRoutes;
```

```
    int numOfBuses;
```

```
public:
```

```
    PublicTransport(string name, int nr, int nor, int nob): Locality(name, nr) {
```

```
        this->numOfRoutes = nor;
```

```
        this->numOfBuses = nob;
```

```
    }
```

```
    void occupanceFactor() {
```

```
        cout << name << endl;
```

```
        int coef;
```

```
        int routeToBus;
```

```
        coef = (numResidents * 0.4) / numOfBuses;
```

```
        routeToBus = numOfBuses / numOfRoutes;
```

```
        if (coef < 400) cout << "Суммарное число автобусов на
```

```
        else if (coef > 400 && coef < 1000) cout << "нагрузка на
```

```
        else cout << "Суммарная высокая нагрузка на
```

```
        транспорт";
```



```
cout << "Среднее кол-во автобусов на одну  
железнодорожную станцию: " << routeBus << "\n\n";
```

```
3;  
3;  
class Village : public Locality {  
private:  
    int distanceToRiver;  
    int distanceToForest;  
public:  
    Village(string name, int nr, int str, int ftt) :  
        Locality(name, nr) {  
        this->distanceToRiver = str;  
        this->distanceToForest = ftt;  
    }  
};
```

```
void setInfo() {  
    cout << "Название села: " << name << endl;  
    cout << "Кол-во жителей: " << numberInhabitants << endl;  
    cout << "Расстояние до реки: " << distanceToRiver << endl;  
    cout << "Расстояние до леса: " << distanceToForest << endl;  
}
```

```
3;  
int main() {  
    setLocate(LC - ALL, "ru");
```

```
    PublicTransport C1("Харьков", 1500000, 140,  
                        1500);
```

```
c3- PublicTransport C2("Киев", 2000000, 200, 1000);
```

```
    Village("Хутор", 500, 1300, 500);
```

```
    C1.occupanceFactor();
```

```
    C2.occupanceFactor();
```

```
    C3.setInfo();
```

```
    return 0;
```

```
3;
```