

Основное задание. Написать параметризованный класс-контейнер, сохраняющий порядок добавления элементов. Класс должен иметь:

- метод, позволяющий добавлять новые элементы;
- итератор, позволяющий перебирать содержимое в порядке их добавления;
- возможность удалять элементы по значению.

Например, если в контейнер были добавлены следующие строки:

```
"aaa", "bbb", "ccc", "aaa", "ddd", "eee", "aaa", "aaa", "fff", "ggg"
```

то после удаления "aaa" должны остаться:

```
"bbb", "ccc", "ddd", "eee", "fff", "ggg"
```

Причём время удаления всех элементов, равных данному, не должно зависеть от общего числа хранимых элементов. Если n – общее число элементов, k – число элементов, подлежащих удалению, а $T(k, n)$ – время удаления, то

$$T(k, n) \leq c * k,$$

где c не зависит от n и k .

Дополнительное задание. Наследование и equals().

Пусть определён базовый класс point, имеющий два свойства getX() и getY():

```
abstract class Point{
    abstract Number getX();
    abstract Number getY();
}
```

а так же определены его наследники:

```
class IntPoint extends Point{
    int x;
    int y;
    Number getX(){ return new Integer(x); }
    Number getY(){ return new Integer(y); }
    IntPoint(int x, int y){
        this.x = x;
        this.y = y;
    }
}
```

```
class DoublePoint extends Point{
    double x;
    double y;

    Number getX() { return new Double(x); }
    Number getY() { return new Double(y); }
    DoublePoint(double x, double y){
        this.x = x;
        this.y = y;
    }
}
```

```
class ColoredPoint extends DoublePoint{
    SystemColor color;
    ColoredPoint(double x, double y, SystemColor color){
        super(x, y);
        this.color = color;
    }
}
```

Переопределите в классах point и coloredPoint метод equals() так, чтобы он удовлетворял условиям, описанным в документации к классу Object (см. документацию)