

## Домашнее задание № 3

### Задача 1

Интерфейс дека (очереди с двумя концами) имеет следующее описание:

```
interface Deque<E> extends Iterable<E> {  
    void addFirst(E elem);    // Добавляет элемент в начало очереди.  
    E first();                // Выдает первый элемент очереди.  
    E removeFirst();          // Удаляет первый элемент из очереди  
                               // (и выдает его).  
    void addLast(E elem);     // Добавляет элемент в конец очереди.  
    E last();                 // Выдает последний элемент очереди.  
    E removeLast();           // Удаляет последний элемент из очереди  
                               // (и выдает его).  
}
```

Надо написать две реализации этого интерфейса - в виде массива элементов (`ArrayDeque`) и в виде списка (`ListDeque`). Программа запускается с параметрами - имя класса реализации и (для реализации в виде массива) максимальный размер дека. Программа должна динамически загрузить класс с нужной реализацией, создать дек из целых чисел, заполнить его несколькими случайными элементами, вычислить сумму элементов дека, используя итератор, а затем проверить, есть ли в загруженной реализации дополнительный реализованный метод `size` (вычисляющий число элементов в очереди). Если такой метод есть, то программа должна его вызвать и вывести общее число добавленных элементов; если нет - сообщить об отсутствии метода в реализации.

### Задача 2

Класс содержит описания методов сортировки массива объектов разными способами. Каждый метод имеет стандартный интерфейс вида

```
public static <E extends Comparable<E>> void sort(E[] array)
```

где `sort` - имя метода. Реализовать три разных метода сортировки (например, “простыми вставками”, “слиянием”, “быстрая”, можно выбрать любые свои, лишь бы интерфейс метода был таким, как заявлено). Написать программу, которая получает в качестве аргумента название метода, создает случайный массив объектов, а затем сортирует указанным методом.