

Задача 5. LRU кеш

Требуется реализовать коллекцию (структуру данных), соответствующую LRU кешу (least recently used). Это структура данных, схожая с ассоциативным массивом (map), но имеющая ограничение на максимальный размер. При записи нового элемента в кеш в первую очередь вытесняется элемент, который не был использован дольше всех.

Интерфейс коллекции LRU кеш

```
/**
 * Коллекция, реализующая LRU кеш.
 *
 * @param <K> Тип ключей
 * @param <V> Тип значений
 */
public interface LruCache<K, V> {

    /**
     * Возвращает значение, соответствующее указанному ключу.
     * При этом элемент (пара ключ-значение) помечается
     * как последний использованный.
     *
     * @param key Ключ
     * @return Значение или {@code null},
     *         если значение не найдено
     */
    V get(K key);

    /**
     * Добавляет элемент (пару ключ-значение) в коллекцию.
     * В случае, если элемент с таким ключом уже был
     * в коллекции, он заменяется.
     * При этом элемент помечается как последний использованный.
     * <p>
     * В случае, если до вставки размер коллекции был равен
     * максимальному, из нее удаляется элемент,
     * неиспользованный дольше всех.
     *
     * @param key Ключ
     * @param value Значение
     */
}
```

```
    */  
    void set(K key, V value);  
  
    /**  
     * Возвращает текущий размер коллекции.  
     *  
     * @return Текущий размер  
     */  
    int getSize();  
  
    /**  
     * Возвращает максимальный размер коллекции.  
     *  
     * @return Максимальный размер  
     */  
    int getLimit();  
  
}
```

Рекомендации по реализации программы

Постарайтесь найти решение, оптимальное с точки зрения алгоритмической сложности операций чтения и записи. Подумайте об использовании двух других, уже известных вам структур данных.