

Bài 1:

Tạo danh sách móc nối (LinkedList) theo mẫu sau.

```
public class SimpleLinkedList<T> {
    class Node {
        T data;
        Node next;
    }
    private Node top = null;
    private Node bot = null;
    private int n = 0;
    public void add(T data) {
        // Thêm phần tử vào đầu danh sách
    }
    public void addBot(T data) {
        // Thêm phần tử vào cuối danh sách
    }
    public T get(int i)
    {
        // Lấy phần tử ở vị trí thứ i
        return null;
    }
    public void set(int i, T data)
    {
        // Gán giá trị ở vị trí i bằng data
    }
    public boolean isContain(T data) {
        // Kiểm tra trong danh sách có chứa phần tử data hay không?
        return false;
    }
    public int size() {
        // Trả lại thông tin số phần tử có trong danh sách
        return 0;
    }
    public boolean isEmpty() {
        // Kiểm tra danh sách có rỗng hay không?
        return true;
    }
    public T removeTop() {
        // Xóa phần tử ở đầu danh sách, trả lại giá trị data của phần tử đó
        return null;
    }

    public T removeBot() {
        // Xóa phần tử ở cuối danh sách, trả lại giá trị data của phần tử đó
        return null;
    }

    public void remove(T data) {
        // Xóa tất cả các phần tử có giá trị bằng data
    }
}
```

Bài 2:

Tạo danh sách (List) bằng kiểu dữ liệu mảng (array) như sau:

Tạo giao diện ListInterface kế thừa giao diện Iterable

```
public interface ListInterface<T> extends Iterable<T>{  
  
    public void add(T data);  
    public T get(int i);  
    public void set(int I, T data);  
    public void remove(T data);  
    public void isContain(T data);  
    public int size();  
    public boolean isEmpty();  
}
```

Tạo lớp SimpleArrayList cài đặt giao diện ListInterface

```
public class SimpleArrayList<T> implements ListInterface<T> {  
    private T[] array;  
    private int n = 0;  
    private int defaultSize = 100;  
  
    public SimpleArrayList() {  
        array = (T[]) new Object[defaultSize];  
    }  
  
    public SimpleArrayList(int capacity) {  
        // Hàm dựng với kích thước mảng là capacity  
    }  
    public void add(T data) {  
    }  
  
    public T get(int i) {  
        return null;  
    }  
    public void set(int i, T data) {  
    }  
  
    public void remove(T data) {  
    }  
  
    public void isContain(T data) {  
    }  
  
    public int size() {  
        return 0;  
    }  
  
    public boolean isEmpty() {  
        return false;  
    }  
    public Iterator<T> iterator() {  
        return null;  
    }  
}
```

Bài 3:

Sử dụng các cấu trúc dữ liệu ở bài 1, 2 để viết chương trình đếm số lần xuất hiện của các từ trong văn bản.

Tạo đối tượng là WordCount gồm 2 thuộc tính là word và count. Đối tượng WordCount nạp chồng phương thức equals(Object o) để có thể sử dụng phương thức isContain đã xây dựng ở các cấu trúc dữ liệu trên, hoặc có thể sử dụng phương thức indexOf của các đối tượng cài đặt giao diện List.