

# Môn học lý thuyết ngôn ngữ hướng đối tượng

## Bài thực hành số 11: Các lớp tập hợp trong Java

**Bài 1)** Xây dựng một chương trình quản lý danh sách các sinh viên của một lớp. Mỗi sinh viên gồm các thông tin: mã số sinh viên (MSSV), họ tên, ngày sinh, giới tính, tên lớp, khóa. Chương trình sẽ thực hiện các chức năng cụ thể như sau:

- Add(student, p): Bổ sung thêm một sinh viên student vào một vị trí p bất kỳ trong danh sách (phần tử đầu tiên có vị trí là 1, và với p=0 tức là bổ sung vào cuối danh sách; Không cho phép 2 sinh viên có cùng MSSV trong danh sách).
- Remove(mssv): Loại bỏ một sinh viên với MSSV mssv cho trước.
- Remove(p): Loại bỏ sinh viên ở vị trí thứ p bất kỳ trong danh sách (phần tử đầu tiên có vị trí là 1).
- Search(s): Tìm một sinh viên theo MSSV hoặc theo tên (với một thông tin s vào thì trước tiên tìm sinh viên có MSSV = s, nếu không thấy thì tìm sinh viên có Họ tên = s).
- Print(): In ra nội dung danh sách sinh viên.
- Save(filename): Lưu nội dung của danh sách sinh viên ra một tệp có tên filename.
- Load(filename): load nội dung danh sách từ một tệp có tên filename.
- Sort(type): sắp xếp danh sách theo MSSV (nếu type=ID) hoặc theo Họ tên (nếu type=NAME).

**Bài 2)** Lớp MyArray được định nghĩa như sau:

```
class MyArray {
    Object[] a;
    public MyArray (int size) {...} // constructor
    public Object at (int i) {...} // access an element
    public void set (int i, Object value) {...}
    class MyIterator implements Iterator {
        // an inner class! ...
        public Iterator iterator () {...}
    }
    ...
}
```

Những đoạn mã dưới đây có hợp lệ không? Nếu hợp lệ thì hãy cho biết kết quả thu được, nếu không hợp lệ hãy giải thích nguyên nhân?

**a)**

```
MyArray a = new MyArray(10);
a.set(0, "hi");
a.set(1, "there");
Iterator i = a.iterator();
System.out.println(i.next());
```

.....

.....

.....

.....

.....

**b)**

```
MyArray a = new MyArray(10);
a.set(0, "hi");
```

```

a.set(1, "there");
MyIterator i = a.iterator();
System.out.println(i.next());

```

```

.....
.....
.....
.....
.....
.....

```

**c)**

```

MyArray a = new MyArray(10);
a.set(0, "hi there");
Iterator i = new MyArray.MyIterator();
System.out.println(i.next());

```

```

.....
.....
.....
.....
.....
.....

```

### Bài 3) Xây dựng chương trình thao tác với HashMap:

```

import java.util.HashMap;
import java.util.Iterator;
import java.util.Map;
import java.util.Set;
public class SimpleCodeCJava {
    public static void main(String[] args) {
        HashMap<Integer, String> hmap = new HashMap<Integer, String>();
        /*Thêm giá trị tương ứng vào các key.*/
        hmap.put(12, "Hà Nội");
        hmap.put(2, "Đà Nẵng");
        hmap.put(7, "Nha Trang");
        hmap.put(49, "Thành Phố Hồ Chí Minh");
        hmap.put(3, "Cà Mau");
        /* Hiện thị giá trị bên trong HashMap sử dụng Iterator*/
        Set set = hmap.entrySet();
        Iterator iterator = set.iterator();
        while(iterator.hasNext()) {
            Map.Entry mentry = (Map.Entry)iterator.next();
            System.out.print("key is: "+ mentry.getKey() + " & Value is: ");
            System.out.println(mentry.getValue());
        }

        /* Lấy ra giá trị dựa vào key*/
        String var= hmap.get(2);
        System.out.println("Value at index 2 is: "+var);

        /* Xóa dữ liệu dựa vào key*/
        hmap.remove(3);
        System.out.println("Map key and values after removal:");
        Set set2 = hmap.entrySet();
        Iterator iterator2 = set2.iterator();
        while(iterator2.hasNext()) {
            Map.Entry mentry2 = (Map.Entry)iterator2.next();
            System.out.print("Key is: "+mentry2.getKey() + " & Value is: ");

```

```
        System.out.println(mentry2.getValue());  
    }  
}
```

**Bài 4)** Viết chương trình *hỗ trợ tra cứu từ điển đơn giản*. Chương trình lưu các từ và loại từ, nghĩa của từ, câu ví dụ,... trong một Collection.