ĐÁP ÁN

Kiểm tra cuối kỳ Đề 2

Câu 1. (2.5 điểm) Chỉ trả lời Đúng hoặc Sai.

a. Các lời gọi phương thức trong Java đều là liên kết động (Sai)

b. Phương thức khởi tạo không thể được khai báo với từ khóa *private* (Sai)

c. Phương thức thực thể có thể truy xuất các thuộc tính lớp (thuộc tính static) (Đúng)

d. Một lớp là trừu tượng thì bắt buộc phải chứa phương thức trừu tượng (Sai)

e. Phương thức khởi tạo của lớp cha sẽ được kế thừa ở lớp con (Sai)

f. Trong cùng một lớp, nếu hai phương thức có cùng tên thì bắt buộc phải khác tham số (Đúng)

g. Giao diện (interface) có thể không cần khai báo phương thức nào (Đúng)

h. Máy ảo Java sẽ biên dịch chương trình nguồn viết bằng ngôn ngữ Java thành bytecode (Sai)

i. Nếu lớp A không định nghĩa bất cứ phương thức khởi tạo nào, A sẽ được trình biên dịch cung cấp phương thức khởi tạo mặc định. (Đúng)

j. Tất cả các lớp trong Java đều có thể có nhiều lớp con (Sai)

Câu 2 (2 điểm)

a. Hãy định nghĩa một lớp ngoại lệ và cho một ví dụ thể hiện cách sử dụng lớp ngoại lệ đấy. (1 điểm)

```
class MyException extends Exception{
}
void test(){
    try{}
    catch(MyException e){...}
}
```

b. Cho một ví dụ và giải thích về việc sử dụng giao diện (interface) làm kiểu của tham số của phương thức. (1 điểm)

```
public interface I1{};

void test(I1 o){...}
```

Trong ví dụ trên, phương thức *test()* sử dụng I1 làm kiểu của tham số. Lúc gọi *test()*, chúng ta cần truyền vào một đối tượng mà lớp của đối tượng đó có cài đặt giao diện I1.

Câu 3 (3 điểm)

Cấu trúc lưu trữ hệ thống file trong máy tính bao gồm: ***Folder, TextFile, BinaryFile***, và ***Shortcut*** gọi chung là các ***Entries***. Khi ***Entry*** là một ***Folder***, bên trong nó lại có thể bao gồm nhiều ***Entries*** khác.

a. Hãy xây dựng (vẽ sơ đồ) thiết kế cho cấu trúc lưu trữ hệ thống file trong máy tính gồm các thành phần như trên.

b. Hãy cài đặt thiết kế trên với phương thức ***visit()*** (hoặc giao diện - interface) để có thể duyệt tất cả các phần tử từ một Entry. Tại mỗi Entry duyệt qua in ra thông tin Entry là *Folder, Textfile, Binaryfile, hay Shortcut*. Nếu Entry là Folder thì duyệt tiếp các phần tử trong Folder đó.

```
class Entry{
        public void visit();
}
class Folder extends Entry{
        ArrayList<Entry> entries;
        public void visit(){
                int n=entries.size();
                for(int i=0;i<n;i++)
                        entries.get(i).visit();
        };
}
class Textfile extends Entry{
        public void vistit(){
                System.out.println("Textfile");
        }
}
```

Câu 4 (2.5 điểm)

a. Định nghĩa lớp A với phương thức *int getNumberofInstance()* trả về số đối tượng thuộc A đã được tạo ra. (1 điểm)

```
public class A{
        private static int n=0;
        public A(){
        n++;
```

```
    }
    public int getNumberofInstance(){// khai báo static cũng được
        return n;
    }
    }
```

b. Định nghĩa lớp tổng quát Group với phương thức *getMax*() trả về đối tượng *lớn nhất* trong Group (Giả sử trong một đối tượng Group chỉ có 3 phần tử). (1 điểm)

```
interface Comparable<T>{
    int compareTo(T o);
}
public class Group<T extends Comparable>{
    T first;
    T second;
    T third;
    public T getMax(){
        T max=first;
        if(second.compareTo(max)>0)
            max=second;
        if(third.compareTo(max)>0)
            max=third;
        return max;
}
}
```

c. Mô tả mẫu thiết kế Singleton (0.5 điểm)

```
class Singleton{
    private static Singleton instance;
    private Singleton(){};
    public static Singleton getInstance(){
        if(instance==null)
            instance=new Singleton();
        return instance;

    }
}
```

Lập trình hướng đối tượng HKI 2015 - 2016