

ĐỀ THI GIỮA KỲ

Môn Cấu trúc dữ liệu và Thuật toán

Thời gian 75 phút

Đề 1

Bài 1. (4 điểm)

Lan đi siêu thị được khuyến mại một số sản phẩm, tuy nhiên do ô tô của Lan không đủ chỗ để chứa hết sản phẩm, nên Lan phải bớt lại 1 loại sản phẩm. Mỗi loại sản phẩm được biểu diễn bằng giá trị của nó là một số nguyên dương (hai loại sản phẩm khác nhau sẽ có giá trị khác nhau). Hãy giúp Lan đưa ra quyết định để tổng số giá trị còn lại sau khi loại bớt 1 loại sản phẩm là lớn nhất..

- Hoàn thiện phương thức `public long getMaxValue(int[] v)` trong lớp `Value`, phương thức này trả lại giá trị là tổng các phần tử còn lại đạt giá trị lớn nhất sau khi loại bớt 1 loại phần tử.
- Ví dụ $v = \{6, 5, 2, 1, 2, 1, 1\}$
- Kết quả thu được là $15 = 6 + 5 + 2 + 2$ (loại các phần tử 1)

Chú ý: sinh viên thiết kế thuật giải cho bài toán trên với độ phức tạp $O(n \log(n))$ với n là số phần tử trong mảng v . Với thuật giải độ phức tạp $O(n^2)$ sẽ không được 100% số điểm của bài này.

Bài 2. (6 điểm)

- Cho giao diện `Queue` đơn giản như sau:

```
public interface QueueInterface<T> {  
    public void enqueue(T element);  
  
    public T dequeue();  
  
    public boolean isEmpty();  
  
}
```

- a) Hoàn thiện các phương thức trong lớp `LinkedListQueue` (3 điểm)

```
public class LinkedListQueue<T> implements QueueInterface<T>{  
  
    class Node {  
        T element;  
        Node next;  
    }
```

```

    }

    @Override
    public void enqueue(T element) {
        // TODO Auto-generated method stub

    }

    @Override
    public T dequeue() {
        // TODO Auto-generated method stub
        return null;
    }

    @Override
    public boolean isEmpty() {
        // TODO Auto-generated method stub
        return false;
    }

}

```

Chú ý: cả 3 phương thức phải hoạt động đúng mới có điểm, nếu 1 trong 3 phương thức hoạt động sai hoặc không hoạt động sẽ được 0 điểm.

- b) Hoàn thiện phương thức void travel(Node tree) trong lớp Tree để in ra các node trong cây theo từng tầng của cây (3 điểm)

```

public class Tree<T> {

    class Node{
        T data;
        Node left;
        Node right;
    }

    void travel(Node tree)
    {

    }

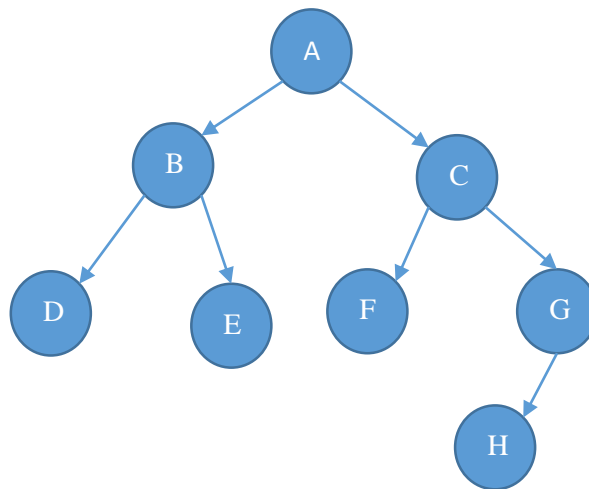
}

```

Gợi ý:

Sử dụng cấu trúc dữ liệu hàng đợi LinkedListQueue đã xây dựng ở ý trên để duyệt cây theo thuật toán sau:

- Đầu tiên đưa nút gốc vào hàng đợi
- Thực hiện các bước sau cho đến khi hàng đợi rỗng
 - o Lấy 1 phần tử **node** ra khỏi hàng đợi, in ra giá trị của node
 - o Đưa các nút con của **node** (nếu có) vào hàng đợi
- Sinh viên không cần quan tâm đến việc xây dựng cây, chỉ cần hoàn thành hàm duyệt cây theo yêu cầu của đề bài.



Ví dụ cây như hình, kết quả là: “A B C D E F G H”

Chú ý:

- Nếu không làm được ý a thì sẽ không được chấm điểm ý này
- Mỗi giá trị của node cách nhau bằng 1 dấu cách

Sinh viên có thể submit bài nhiều lần, bài submit lần cuối sẽ là bài nộp của sinh viên.

Bài làm của sinh viên có thể được đánh giá trên một bộ test mẫu, tuy nhiên đây không phải là điểm cuối cùng, bài làm sẽ được chấm lại trên một bộ test khác và thông báo cho sinh viên sau.

Bài làm của sinh viên phải chạy được, theo đúng yêu cầu đề bài, các bài làm có lỗi biên dịch, cú pháp,.. chương trình không chạy sẽ được 0 điểm.

Các hình thức vi phạm quy chế thi như sử dụng USB, tài liệu, điện thoại, làm bài hộ, thi hộ,... sẽ bị 0 điểm cả môn học.