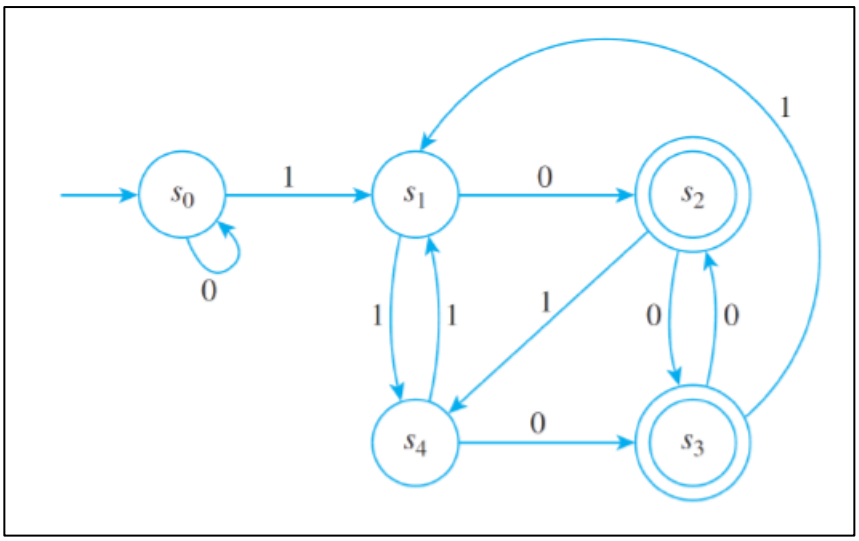
```java
public class Main {
    public static void main(String[] args) {
        EventualState Test = new EventualState(5);
        Test.setStarts(0);
        Test.setFinals(2,4);
        Test.setNode("0,1 2,4 3,4 2,1 3,1");
        Test.Input(0,1,1,1,0,0,0,1,0,1,0);
    }
}
```




```java
class node{
    private  int zero;
    private  int one;
    node(int ze, int on){
        zero = ze;
        one = on;
    }
    public int getZero() {
        return zero;
    }
    public int getOne() {
        return one;
    }
}
```

```
[0]--0->[0]--1->[1]--1->[4]--1->[1]--0->[2]--0->[3]--0->[2]--1->[4]--0->[3]--1->[1]--0->[2]
Accept
```

```java
class EventualState{
    private node[] list;
    private int starts;
    private int[] finals;

    EventualState(int l){
        list = new node[l];
    }

    void setNode(String str){
        String[] st = str.split(" ");
        for(int i=0; i<st.length; i++){
            list[i] = new node(Integer.parseInt(st[i].split(",")[0]),
                               Integer.parseInt(st[i].split(",")[1]));
        }
    }

    void setStarts(int starts) {
        this.starts = starts;
    }

    void setFinals(int... finals) {
        this.finals = finals;
    }

    void Input(int... input){
        int now = starts;
        System.out.print("["+now+"]");
        for(int i : input){
            if(i == 0){
                 now = list[now].getZero();
                System.out.print("--0->["+now+"]");
            }else{
                now = list[now].getOne();
                System.out.print("--1->["+now+"]");
            }
        }
        System.out.println();
        if(isfinals(now)){
            System.out.println("Accept");
        }else{
            System.out.println("Reject");
        }
    }

    private boolean isfinals(int n){
        for(int f:finals)if(n==f)return true;
        return false;
    }
}
```