*** EXPERIMENT NO: 05 ***

Author : Pradyot Patil

Roll No : 53 [7B]

Date : 21-SEP-2018

```
package parser;
import java.util.* ;
/**
 * @author Paddi
 */
public class Parser {
   static production pr[]=new production[10];
   static String buffer="$ championhip the won India" ;
   static String stack="S $" ;
   static String
str[]={"championship","ball","toss","is","want","won","played","me","I","you","India","Australia"} ;

   public static void main(String[] args) {
      input() ;
      String f[]=buffer.split(" ");
      int flag=0 ;
      while(buffer.compareTo(stack)!=0)
      {
         if(iterate()!=1)
         {
            flag=1 ;
            break ;
         }
      }
      if(flag==1)
```

```
        {
              System.out.println("String Invalid") ;
        }
        else
           System.out.println("String Valid") ;
    }
static int iterate()
{
    String f[]=buffer.split(" ");
    String d[]=stack.split(" ") ;
    int flag=1 ;
    if(f[f.length-1].compareTo(d[0])==0)
    {
        buffer="" ;
        for(int i=0;i<f.length-1;i++)
        {
           buffer=buffer+" "+f[i] ;
        }
        buffer=buffer.substring(1, buffer.length()) ;
        stack="" ;
        for(int i=1;i<d.length;i++)
        {
           stack=stack+" "+d[i] ;
        }
        stack=stack.substring(1, stack.length()) ;
        flag=0;
    }
    else
    {
        String last=f[f.length-1] ;
        String first=d[0] ;
```

```
// System.out.println(first);
 for(int i=0;i<8;i++)
 {
    if(pr[i].parent.compareTo(first)==0)
    {
      ArrayList<element> temp=pr[i].row ;
       for(element e : temp)
       {
          if(e.terminal.compareTo(last)==0)
          {
            // System.out.println("yo");
             flag=0 ;
             ArrayList<String> ars=e.arr ;
             String s="" ;
             for(String t : ars)
             {
                s=s+" "+t ;
             }
             stack=s ;
             for(int j=1;j<d.length;j++)
             {
                stack=stack+" "+d[j] ;
             }
             stack=stack.substring(1, stack.length()) ;
             break ;
          }
       }
       if(flag==1)
       {
          int max=0 ;
          int c=-1 ;
```

```
                for(int k=0;k<str.length;k++)
                {
                   int b=Longest_common_substr.printLCSubStr(last,str[k],last.length(),str[k].length()) ;
                   if(b>max)
                   {
                      max=b ;
                      c=k ;
                   }
                }
                System.out.println("Did you mean "+str[c]+" instead of "+last);
             }
            break ;
           }
        }
    }
    if(flag==0)
    {
       System.out.println("buffer: ("+buffer+" )");
       System.out.println("stack: ("+stack+" )");
       return 1 ;
    }
    return 0 ;
}
static void input()
{
    ArrayList<element> r1=new ArrayList() ;
    r1.add(new element("is","NP VP"));
    r1.add(new element("want","NP VP"));
    r1.add(new element("won","NP VP"));
    r1.add(new element("played","NP VP"));
    r1.add(new element("me","NP VP"));
```

```
r1.add(new element("I","NP VP"));
r1.add(new element("you","NP VP"));
r1.add(new element("India","NP VP"));
r1.add(new element("Australia","NP VP"));
r1.add(new element("Steve","NP VP"));
r1.add(new element("John","NP VP"));
r1.add(new element("the","NP VP"));
r1.add(new element("a","NP VP"));
r1.add(new element("an","NP VP"));
pr[0]=new production("S",r1);
ArrayList<element> r2=new ArrayList() ;
r2.add(new element("me","P")) ;
r2.add(new element("I","P")) ;
r2.add(new element("you","P")) ;
r2.add(new element("India","PN")) ;
r2.add(new element("Australia","PN")) ;
r2.add(new element("Steve","PN")) ;
r2.add(new element("John","PN")) ;
r2.add(new element("the","D N")) ;
r2.add(new element("a","D N")) ;
r2.add(new element("an","D N")) ;
pr[1]=new production("NP",r2);
ArrayList<element> r3=new ArrayList() ;
r3.add(new element("is","V NP")) ;
r3.add(new element("want","V NP")) ;
r3.add(new element("won","V NP")) ;
r3.add(new element("palyed","NP")) ;
pr[2]=new production("VP",r3);
ArrayList<element> r4=new ArrayList() ;
r4.add(new element("championship","championship")) ;
r4.add(new element("ball","ball")) ;
```

```
    r4.add(new element("toss","toss")) ;
    pr[3]=new production("N",r4);
    ArrayList<element> r5=new ArrayList() ;
    r5.add(new element("is","is")) ;
    r5.add(new element("want","want")) ;
    r5.add(new element("won","won")) ;
    r5.add(new element("played","played")) ;
    pr[4]=new production("V",r5);
    ArrayList<element> r6=new ArrayList() ;
    r6.add(new element("me","me")) ;
    r6.add(new element("I","I")) ;
    r6.add(new element("you","you")) ;
    pr[5]=new production("P",r6);
    ArrayList<element> r7=new ArrayList() ;
    r7.add(new element("India","India")) ;
    r7.add(new element("Australia","Australia")) ;
    r7.add(new element("Steve","Steve")) ;
    r7.add(new element("John","John")) ;
    pr[6]=new production("PN",r7);
    ArrayList<element> r8=new ArrayList() ;
    r8.add(new element("the","the")) ;
    r8.add(new element("a","a")) ;
    r8.add(new element("an","an")) ;
    pr[7]=new production("D",r8);
}
}

class element{
    String terminal ;
    ArrayList<String> arr ;
    element(String ter,String prod)
```

```
    {
        this.terminal=ter ;
        arr=new ArrayList() ;
        String s[]=prod.split(" ");
        for(int i=0;i<s.length;i++)
                        arr.add(s[i]) ;

    }
}
class production{
    String parent ;
    ArrayList<element> row ;
    production(String s,ArrayList<element> r)
    {
        this.parent=s ;
        this.row=r ;
    }
}
 class Longest_common_substr {
    /* function to find and print the longest common substring of X[0..m-1] and Y[0..n-1] */
    static int printLCSubStr(String X, String Y, int m, int n)
    {
        int[][] LCSuff = new int[m + 1][n + 1];
        // To store length of the longest common substring
        int len = 0;


        int row = 0, col = 0;
              for (int i = 0; i <= m; i++) {
           for (int j = 0; j <= n; j++) {
              if (i == 0 || j == 0)
```

```
                LCSuff[i][j] = 0;
            else if (X.charAt(i - 1) == Y.charAt(j - 1)) {
                LCSuff[i][j] = LCSuff[i - 1][j - 1] + 1;
                if (len < LCSuff[i][j]) {
                    len = LCSuff[i][j];
                    row = i;
                    col = j;
                }
            }
            else
                LCSuff[i][j] = 0;
        }
    }
    if (len == 0) {
        System.out.println("No Common Substring");
        return 0;
    }
            String resultStr = "";
    while (LCSuff[row][col] != 0) {
        resultStr = X.charAt(row - 1) + resultStr; // or Y[col-1]
        --len;
        // move diagonally up to previous cell
        row--;
        col--;
    }
    // required longest common substring
    return resultStr.length() ;
//System.out.println(resultStr);
   }
}
/*OUTPUT
```

Input: championship the won India

buffer: ($ championship the won India)

stack: (NP VP $)

buffer: ($ championship the won India)

stack: (PN VP $)

buffer: ($ championship the won India)

stack: (India VP $)

buffer: ($ championship the won)

stack: (VP $)

buffer: ($ championship the won)

stack: (V NP $)

buffer: ($ championship the won)

stack: (won NP $)

buffer: ($ championship the)

stack: (NP $)

buffer: ($ championship the)

stack: (D N $)

buffer: ($ championship the)

stack: (the N $)

buffer: ($ championship)

stack: (N $)

buffer: ($ championship)

stack: (championship $)

buffer: ($)

stack: ($)

String Valid

Input: championhip the won India

Did you mean championship instead of championhip

String Invalid

*/