

组成最大数题目描述：

小组中每位都有一张卡片，卡片上是 6 位内的正整数，将卡片连起来可以组成多种数字，计算组成的最大数字。

输入描述：

“,”号分割的多个正整数字符串，不需要考虑非数字异常情况，小组最多 25 个人

输出描述：

最大的数字字符串

补充说明

示例 1

输入：

22,221

输出：

22221

说明：

示例 2

输入：

4589,101,41425,9999

输出：

9999458941425101

说明：

```
import java.util.Scanner;
```

```
// 注意类名必须为 Main, 不要有任何 package xxx 信息
```

```
public class Main {  
  
    public static void main(String[] args) {  
  
        Scanner in = new Scanner(System.in);  
  
        if(in.hasNextLine())  
  
        {  
  
            String str=in.nextLine();  
  
            String[] card=str.split(",");  
  
            boolean[] index= new boolean[card.length];  
  
            String result="";  
  
            for(int i=0;i<card.length;i++)  
  
            {  
  
                int temp=-1;  
  
                for(int j=0;j<card.length;j++)  
  
                {  
  
                    if(index[j]==true)  
  
                        continue;  
  
                    if(temp<0||bijiao(card[j],card[temp]))  
  
                        temp=j;  
  
                }  
  
                index[temp]=true;  
  
                result+=card[temp];  
  
            }  
  
        }  
  
    }  
  
}
```

```

        System.out.println(result);

    }

}

public static boolean bijiao( String a,String b){

    int length=Math.min(a.length(),b.length());

    for(int i=0;i<length;i++)

    {

        if(a.charAt(i)<b.charAt(i))

        {

            return false;

        }

        else if(a.charAt(i)>b.charAt(i))

        {

            return true;

        }

    }

    if(a.length()>length)

    {

        return a.charAt(length)>b.charAt(0);

    }

    if(b.length()>length)

```

```
{  
    return b.charAt(length)<a.charAt(0);  
}  
  
return true;  
}  
}
```