

## 程序设计实训 练习3

# 乒乓球

## 描述

【问题背景】国际乒联现在主席沙拉拉自从上任以来就立志于推行一系列改革，以推动乒乓球运动在全球的普及。其中11分制改革引起了很大的争议，有一部分球员因为无法适应新规则只能选择退役。华华就是其中一位，他退役之后走上了乒乓球研究工作，意图弄明白11分制和21分制对选手的不同影响。在开展他的研究之前，他首先需要对他多年比赛的统计数据进行一些分析，所以需要你的帮忙。

【问题描述】华华通过以下方式进行分析，首先将比赛每个球的胜负列成一张表，然后分别计算在11分制和21分制下，双方的比赛结果（截至记录末尾）。

比如现在有这么一份记录，（其中W表示华华获得一分，L表示华华对手获得一分）：

WWWWWWWWWWWWWWWWWWWWWWWWLW

在11分制下，此时比赛的结果是华华第一局11比0获胜，第二局11比0获胜，正在进行第三局，当前比分1比1。而在21分制下，此时比赛结果是华华第一局21比0获胜，正在进行第二局，比分2比1。如果一局比赛刚开始，则此时比分为0比0。

你的程序就是要对于一系列比赛信息的输入（WL形式），输出正确的结果。

## 输入

每个输入包含若干行字符串（每行至多20个字母），字符串有大写的W、L和E组成。其中E表示比赛信息结束，程序应该忽略E之后的所有内容。

## 输出

输出由两部分组成，每部分有若干行，每一行对应一局比赛的比分（按比赛信息输入顺序）。其中第一部分是11分制下的结果，第二部分是21分制下的结果，两部分之间由一个空行分隔。

### 输入样例 1

```
WWWWWWWWWWWWWWWWWWWWWWWWLW
```

### 输出样例 1

```
11:0
11:0
1:1

21:0
2:1
```

```

#include<iostream>
using namespace std;
int s[62501]; int len;
void func(int x){
    int w=0,l=0;
    for(int i=1;i<=len;i++){
        if(s[i])w++;
        else l++;
        if(w-l>=2&&w>=x){
            printf("%d:%d\n",w,l);
            w=l=0;
        }
        if(l-w>=2&&l>=x){
            printf("%d:%d\n",w,l);
            w=l=0;
        }
    }
    printf("%d:%d\n",w,l);
}
int main(){
    for(char c;cin>>c&&c!='E';)s[++len]=(c=='W'?1:0);
    func(11);
    printf("\n");
    func(21);
    return 0;
}

```

BY 162240216 肖正阳

## 导弹拦截

### 描述

某国为了防御敌国的导弹袭击，开发出一种导弹拦截系统，但是这种拦截系统有一个缺陷：虽然它的第一发炮弹能够到达任意的高度，但是以后每一发炮弹都不能高于前一发的高度。某天，雷达捕捉到敌国的导弹来袭，由于该系统还在试用阶段。所以一套系统有可能不能拦截所有的导弹。

输入导弹依次飞来的高度（雷达给出的高度不大于30000的正整数）。计算要拦截所有导弹最小需要配备多少套这种导弹拦截系统。

### 输入

$n$ 颗依次飞来的高度 ( $1 \leq n \leq 1000$ ) 。

### 输出

要拦截所有导弹最少需要配备的系统数量 $k$ 。

#### 输入样例 1

389 207 155 300 299 170 158 65

#### 输出样例 1

2

### 【算法分析】

按照题意，被一套系统拦截的所有导弹中，最后一枚导弹的高度最低。设：

$k$ 为当前配备的系统数；

$l[k]$ 为被第 $k$ 套系统拦截的最后一枚导弹的高度，简称系统 $k$ 的最低高度（ $1 \leq k \leq n$ ）。

我们首先设导弹1被系统1所拦截（ $k \leftarrow 1, l[k] \leftarrow$ 导弹1的高度）。然后依次分析导弹2, ..., 导弹 $n$ 的高度。

若导弹 $i$ 的高度高于所有系统此前已拦截的最低高度，则断定导弹 $i$ 不能被已有的这些系统所拦截，应增设一套系统来拦截导弹 $i$ （此时 $k \leftarrow k+1, l[k] \leftarrow$ 导弹 $i$ 的高度）；若导弹 $i$ 低于某些系统此前已拦截的最低高度，那么导弹 $i$ 均可被这些系统所拦截。究竟选择哪个系统拦截可使得配备的系统数最少，我们不妨采用贪心策略，选择其中最低高度最小（即导弹 $i$ 的高度与系统已拦截的最低高度最接近）的一套系统 $p$ （ $l[p] = \min\{l[j] | l[j] > \text{导弹}i\text{的高度}\}$ ； $l[p] \leftarrow$ 导弹 $i$ 的高度）（ $1 \leq j \leq k$ ）。这样可使得一套系统拦截的导弹数尽可能增多。

依次类推，直至分析了 $n$ 枚导弹的高度为止。此时得出的 $k$ 便为应配备的最少系统数。参考程序主要框架如下：

```
k=1; l[k]=导弹1的高度;
```

```
for (i=2; i<=n; ++i)
```

```
{   p=0;
```

```
    for (j=1; j<=k; ++j)
```

```
        if (l[j]>=导弹i的高度) { if (p==0) p=j;
```

```
                                else if (l[j]<l[p]) p=j;} //贪心
```

```
    if (p==0) { ++k; l[k]=导弹i的高度; } //增加一套新系统
```

```
    else l[p]=导弹i的高度; //贪心,更新第p套系统的最低高度
```

```
}
```

输出应配备的最少系统数 $K$ 。

```
using namespace std;  
const int N = 1001;
```

```
int n;int q[N];int g[N];
```

```
int main()
```

```
{
```

```
    while (cin >> q[n])
```

```
        n++;
```

```
    int cnt = 0;
```

```
    for (int i = 0; i < n; i++)
```

```
    {
```

```
        int k = 0;
```

```
        while (k < cnt && g[k] < q[i])
```

```
            k++;
```

```
        g[k] = q[i];
```

```
        if (k >= cnt)
```

```
            cnt++;
```

```
    }
```

```
    cout << cnt << endl;
```

```
    return 0;
```

```
}
```

BY 1622162230217 陈梓鹏

## 昆虫繁殖

### 描述

科学家在热带森林中发现了一种特殊的昆虫，这种昆虫的繁殖能力很强。每对成虫过x个月产y对卵，每对卵要过两个月长成成虫。假设每个成虫不死，第一个月只有一对成虫，且卵长成成虫后的第一个月不产卵(过X个月产卵)，问过Z个月以后，共有成虫多少对？

$0 \leq X \leq 20, 1 \leq Y \leq 20, X \leq Z \leq 50$

### 输入

x,y,z的数值

### 输出

过Z个月以后，共有成虫对数

#### 输入样例 1

1 2 8

#### 输出样例 1

37

```
const int maxn=100+5;
int cnt;
long long a[maxn];
int i,j;
```

BY 162210107 蔡蕾

```
int main()
{
    int x,y,z;
    cin>>x>>y>>z;

    for(i=1;i<=x;i++)//产卵之前，都是一个
        a[i]=1;

    for(i;i<=z+1;i++)
        a[i]=a[i-1]+a[i-x-2]*y; //前一月的成虫+ (i-x-2) 个月之前出生的虫子（此时刚刚变成成虫）

    cout<<a[z+1];

    return 0;
}
```