

## 限时测试【2.8】

### 任务：

时间：下午 2.到 5.，完成下面 6 道题

提交：将 c++代码文件私聊发给姐，姐收到之后马上给你们测试，然后告诉你们测试情况和分数

测试情况：

WA(wrong answer)、TLE(time limit error)、

RE(runtime error)、AC(accept)

分数：通过多少个测试样例，就有多少分哦

这次的题目依然是复习为主

除了常见的知识点，最后还来了一道博弈论

博弈论没啥特别需要学的知识点（或者说又偏又难哦

就当一道模拟题玩玩看就好啦

昨天休息了一整天，现在应该精力充沛了叭

So 要习惯三个小时的训练哈

题目	难度	知识点
Playing Small Q	普及/提高-	字符串处理
Playing Deep Tree	普及/提高-	DFS/BFS
Playing Weights	普及-	DP/背包
Playing Roads	普及/提高-	图论
Playing Cows	普及/提高-	图论
Playing Alice And Bob	普及/提高-	模拟/博弈论

题目：

1、Playing Small Q

题目描述 展开

小 Q 给你了几行字符串，字符串中间包含有一些整数，整数之间可能会有其他字符。你需要输出每行中所有整数的和。如果一个行里面一个数都没有则不要输出。**每两个整数之间一定有间隔字符（见样例 3）。**

请注意，题目中的整数都是简单的整数，即：整数前面没有多余的符号。假如数据中有 `.`，视作分隔符。

输入格式

每行一个字符串。

输出格式

对于每一个包含了整数的字符串，输出一行一个数，即其中数的和。

输入输出样例

<p>输入 #1</p> <pre>su57jdkjth54hjsns-321d 8 ejre erg(&amp;~% weruy4uhnd-</pre>	<p>输出 #1</p> <pre>-202 4</pre>
<p>输入 #2</p> <pre>4532984      d      -1</pre>	<p>输出 #2</p> <pre>4532983</pre>
<p>输入 #3</p> <pre>1-1 1 -1 1--5 1-----1 -1-1 --1 -2. 3</pre>	<p>输出 #3</p> <pre>2 0 -4 0 0 -1 1</pre>

说明/提示

- 【样例解释】
- 第一个样例中，第一行有 57, 54, -321, 8 几个数，和为 -202；第二、三行没有数字；第四行只有一个 4，输出 4。
  - 第二个样例中只有一行，有两个数，和为 4532893。
  - 第三个样例的第一行，由于每两个整数之间一定有间隔字符，所以把 `-` 视为间隔字符，输出 2。第五行，第一个数为 -1，因此把第二个 `-` 视为间隔字符，和为  $-1 + 1 = 0$ 。第七行，将 `.` 视为间隔字符，和为  $-2 + 3 = 1$ 。

## 2、Playing Deep Tree

### 题目描述

[展开](#)

给出每个节点的两个儿子节点，建立一棵二叉树，如果是叶子节点，则输入 `0 0`。建好树后希望知道这棵二叉树的深度。二叉树的**深度**是指从根节点到叶子结点时，最多经过了几层。

最多有  $10^6$  个结点。

### 输入格式

无

### 输出格式

无

### 输入输出样例

输入 #1

[复制](#)

输出 #1

[复制](#)

```
7
2 7
3 6
4 5
0 0
0 0
0 0
0 0
```

```
4
```

### 3、Playing Weights

#### 题目描述

[展开](#)

设有 $1g$ 、 $2g$ 、 $3g$ 、 $5g$ 、 $10g$ 、 $20g$ 的砝码各若干枚（其总重 $\leq 1000$ ），

#### 输入格式

输入方式： $a_1, a_2, a_3, a_4, a_5, a_6$

（表示 $1g$ 砝码有 $a_1$ 个， $2g$ 砝码有 $a_2$ 个，...， $20g$ 砝码有 $a_6$ 个）

#### 输出格式

输出方式： $Total = N$

（ $N$ 表示用这些砝码能称出的不同重量的个数，但不包括一个砝码也不用的情况）

#### 输入输出样例

输入 #1

[复制](#)

输出 #1

[复制](#)

1 1 0 0 0 0

Total=3

## 4、Playing Roads

### 题目背景

[展开](#)

A地区在地震过后，连接所有村庄的公路都造成了损坏而无法通车。政府派人修复这些公路。

### 题目描述

给出A地区的村庄数 $N$ ，和公路数 $M$ ，公路是双向的。并告诉你每条公路的连着哪两个村庄，并告诉你什么时候能修完这条公路。问最早什么时候任意两个村庄能够通车，即最早什么时候任意两条村庄都存在至少一条修复完成的道路（可以由多条公路连成一条道路）

### 输入格式

第1行两个正整数 $N, M$

下面 $M$ 行，每行3个正整数 $x, y, t$ ，告诉你这条公路连着 $x, y$ 两个村庄，在时间 $t$ 时能修复完成这条公路。

### 输出格式

如果全部公路修复完毕仍然存在两个村庄无法通车，则输出-1，否则输出最早什么时候任意两个村庄能够通车。

### 输入输出样例

输入 #1

[复制](#)

```
4 4
1 2 6
1 3 4
1 4 5
4 2 3
```

输出 #1

[复制](#)

```
5
```

## 5、Playing Cows

### 题目描述

[展开](#)

John 的农场在给奶牛挤奶前有很多杂务要完成，每一项杂务都需要一定的时间来完成它。比如：他们要将奶牛集合起来，将他们赶进牛棚，为奶牛清洗乳房以及一些其它工作。尽早将所有杂务完成是必要的，因为这样才有更多时间挤出更多的牛奶。当然，有些杂务必须在另一些杂务完成的情况下才能进行。比如：只有将奶牛赶进牛棚才能开始为它清洗乳房，还有在未给奶牛清洗乳房之前不能挤奶。我们把这些工作称为完成本职工作的准备工作。至少有一项杂务不要求有准备工作，这个可以最早着手完成的工作，标记为杂务1。John 有需要完成的 $n$ 个杂务的清单，并且这份清单是有一定顺序的，杂务 $k(k > 1)$ 的准备工作只可能在杂务1至 $k - 1$ 中。

写一个程序从1到 $n$ 读入每个杂务的工作说明。计算出所有杂务都被完成的最短时间。当然互相没有关系的杂务可以同时工作，并且，你可以假定 John 的农场有足够多的工人来同时完成任意多项任务。

### 输入格式

第1行：一个整数 $n$ ，必须完成的杂务的数目( $3 \leq n \leq 10,000$ )；

第2至 $(n + 1)$ 行：共有 $n$ 行，每行有一些用1个空格隔开的整数，分别表示：

\* 工作序号(1至 $n$ ,在输入文件中是有序的)；

\* 完成工作所需要的时间 $len(1 \leq len \leq 100)$ ；

\* 一些必须完成的准备工作，总数不超过100个，由一个数字0结束。有些杂务没有需要准备的工作只描述一个单独的0，整个输入文件中不会出现多余的空格。

### 输出格式

一个整数，表示完成所有杂务所需的最短时间。

### 输入输出样例

输入 #1

[复制](#)

```
7
1 5 0
2 2 1 0
3 3 2 0
4 6 1 0
5 1 2 4 0
6 8 2 4 0
7 4 3 5 6 0
```

输出 #1

[复制](#)

```
23
```

## 6、Playing Alice And Bob

### 题目描述

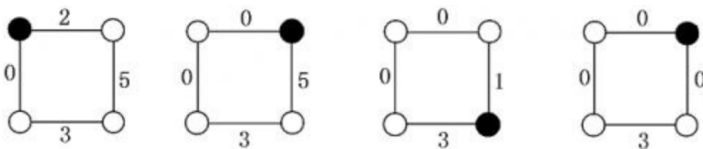
[展开](#)

有一个取数的游戏。初始时，给出一个环，环上的每条边上都有一个非负整数。这些整数中至少有一个0。然后，将一枚硬币放在环上的一个节点上。两个玩家就是以这个放硬币的节点为起点开始这个游戏，两人轮流取数，取数的规则如下：

- (1) 选择硬币左边或者右边的一条边，并且边上的数非0；
- (2) 将这条边上的数减至任意一个非负整数(至少要有所减小)；
- (3) 将硬币移至边的另一端。

如果轮到一个玩家走，这时硬币左右两边的边上的数值都是0，那么这个玩家就输了。

如下图，描述的是Alice和Bob两人的对弈过程，其中黑色节点表示硬币所在节点。结果图(d)中，轮到Bob走时，硬币两边的边上都是0，所以Alcie获胜。



(a) Alice (b) Bob (c) Alice (d) Bob

现在，你的任务就是根据给出的环、边上的数值以及起点（硬币所在位置），判断先走方是否有必胜的策略。

### 输入格式

第一行一个整数N ( $N \leq 20$ )，表示环上的节点数。

第二行N个数，数值不超过30，依次表示N条边上的数值。硬币的起始位置在第一条边与最后一条边之间的节点上。

### 输出格式

仅一行。若存在必胜策略，则输出“YES”，否则输出“NO”。

### 输入输出样例

输入 #1

[复制](#)

输出 #1

[复制](#)

```
4
2 5 3 0
```

YES

输入 #2

[复制](#)

输出 #2

[复制](#)

```
3
0 0 0
```

NO