

青少年软件编程（C 语言）等级考试试卷（四级）

分数：100 题数：4

一、编程题(共 4 题，共 100 分)

1. 最佳路径

如下所示的由正整数数字构成的三角形：

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

从三角形的顶部到底部有很多条不同的路径。对于每条路径，把路径上面的数加起来可以得到一个和，和最大的路径称为最佳路径。你的任务就是求出最佳路径上的数字之和。注意：路径上的每一步只能从一个数走到下一层上和它最近的下边（正下方）的数或者右边（右下方）的数。

时间限制：1000

内存限制：65536

输入

第一行为三角形高度 $100 \geq h \geq 1$ ，同时也是最底层边的数字的数目。从第二行开始，每行为三角形相应行的数字，中间用空格分隔。

输出

最佳路径的长度数值。

样例输入

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

样例输出

```
30
```

提示

如何采用动态规划的思想，对问题进行分解。

试题编号：20230318-4-01

试题类型：编程题

标准答案：

试题难度：一般

试题解析：

展示地址：点击浏览

考生答案：（此题已作答）

考生得分：25

是否评分：已评分

评价描述：

考生答案：

```
#include<bits/stdc++.h>
using namespace std;
const int maxn = 1e2 + 10;
int dp[maxn][maxn], a[maxn][maxn];
int main(){
    int n;
    cin>>n;
    for(int i = 1;i <= n;i++){
        for(int j = 1;j <= i;j++){
            cin>>a[i][j];
        }
        memset(dp,0,sizeof(dp));
        dp[1][1] = a[1][1];
        for(int i = 2;i <= n;i++){
            for(int j = 1;j <= i;j++){
                dp[i][j] = max(dp[i-1][j-1],dp[i-1][j])+a[i][j];
            }
        }
        int m = dp[n][1];
        for(int i = 2;i < n; i++){
            m = max(m,dp[n][i]);
        }
        cout<<m<<endl;
    }
}
```

2. 邮票收集

小A是个邮票收集爱好家，他有 n 种面值的邮票，每种邮票都有无数张。一天小B想要寄信，需要一共面值和为 k 的邮票组合。小A想要知道拼出面值为 k 的邮票最少需要多少张。

时间限制：1000

内存限制：131072

输入

输入是多组数据。(不超过 10 组) 每组数据的第一行正整数 n, k ，表示邮票的种类数目和目标要拼出的钱。 $(0 < n \leq 100, 0 < k \leq 1000)$ 接下来的一行有 n 个正整数 $a_i (0 < a_i \leq 1000)$ 。若 $n=k=0$ 表示输入结束。

输出

每组数据输出一行一个数，分别表示拼出 k 需要的最少的邮票数量。 如果不存在能够拼出 k 的方案，输出 -1。

样例输入

```
4 10
1 2 3 4
5 16
1 2 3 4 5
2 7
4 5
0 0
```

样例输出

```
3
4
-1
```

提示

第一组数据: $10 = 4+4+2$ 第二组数据: $16 = 5+5+5+1$ 第三组数据: 不存在。

试题编号: 20230318-4-02

试题类型: 编程题

标准答案:

试题难度: 一般

试题解析:

展示地址: [点击浏览](#)

考生答案: (此题已作答)

考生得分: 25

是否评分: 已评分

评价描述:

考生答案:

```
#include <bits/stdc++.h>
```

```
using namespace std;
```

```
const int N = 10005;
```

```

const int INF = 0x3f3f3f3f;
int dp[N];
int w[N];
int t = 11 ,n ,m;
int main(){
    while(t--){
        cin >> n >> m;
        if(n == 0)break;
        for(int i=1; i<=n; i++)
            cin >> w[i];
        memset(dp,INF,sizeof(dp));
        dp[0] = 0;
        for(int i = 1; i <= n; i++)
            for(int j = w[i]; j <= m; j++)
                dp[j] = min(dp[j] , dp[j-w[i]] + 1);
        if(dp[m] != INF)
            printf("%d\n",dp[m]);
        else
            printf("-1\n");
    }
    return 0;
}

```

3. 切割回文

阿福最近对回文串产生了非常浓厚的兴趣。

如果一个字符串从左往右看和从右往左看完全相同的话，那么就认为这个串是一个回文串。例如，“abcaacba”是一个回文串，“abcaaba”则不是一个回文串。

阿福现在强迫症发作，看到什么字符串都想要把它变成回文的。阿福可以通过切割字符串，使得切割完之后得到的子串都是回文的。

现在阿福想知道他最少切割多少次就可以达到目的。例如，对于字符串“abaacca”，最少切割一次，就可以得到“aba”和“acca”这两个回文子串。

时间限制：1000

内存限制：65536

输入

输入的第一行是一个整数 T ($T \leq 20$)，表示一共有 T 组数据。接下来的 T 行，每一行都包含了一个长度不超过的 1000 的字符串，且字符串只包含了小写字母。

输出

对于每组数据，输出一行。该行包含一个整数，表示阿福最少切割的次数，使得切割完得到的子串都是回文的。

样例输入

```
3
abaacca
abcd
abcba
```

样例输出

```
1
3
0
```

提示

对于第一组样例，阿福最少切割 1 次，将原串切割为“aba”和“acca”两个回文子串。对于第二组样例，阿福最少切割 3 次，将原串切割为“a”、“b”、“c”、“d”这四个回文子串。对于第三组样例，阿福不需要切割，原串本身就是一个回文串。

试题编号：20230318-4-03

试题类型：编程题

标准答案：

试题难度：一般

试题解析：

展示地址：[点击浏览](#)

考生答案：（此题已作答）

考生得分：0

是否评分：已评分

评价描述：

4. 小球放盒子

有 N 个相同的球， M 个不同的盒子，每个盒子最多放 K 个球

请计算将这 N 个球全部放入盒子中的方案数模 1000007 后的结果

时间限制：10000

内存限制：131072

输入

三个正整数，依次为 N，M，K

输出

输出方案数模 1000007 后的结果

样例输入

```
4 2 3
```

样例输出

```
3
```

提示

总共有 3 种方案，依次为 { 3 ， 1 }，{ 2 ， 2 }，{ 1 ， 3 }。对于 100%的数据， $N, M \leq 5000$

试题编号：20230318-4-04

试题类型：编程题

标准答案：

试题难度：一般

试题解析：

展示地址：[点击浏览](#)

考生答案：（此题已作答）

考生得分：0

是否评分：已评分

评价描述：