

# 全国青少年信息学联赛

## CCF NOIP 2018

题目名称	图的遍历	EASY	字符串匹配
题目类型	传统型	传统型	传统型
目录	dfs	easy	string
可执行文件名	dfs	easy	string
输入文件名	dfs.in	easy.in	string.in
输出文件名	dfs.out	easy.out	string.out
每个测试点时限	1.0 秒	3.0 秒	1.0 秒
内存限制	512MB	512MB	512MB
测试点(包)数目	10	20	20
测试点(包)分值	10	5	5

### 提交源程序文件名

对于 C++语言	dfs.cpp	easy.cpp	string.cpp
对于 C 语言	dfs.c	easy.c	string.c
对于 Pascal 语言	dfs.pas	easy.pas	string.pas

### 编译选项

对于 C++语言	-lm -O2	-lm -O2	-lm -O2
对于 C 语言	-lm -O2	-lm -O2	-lm -O2
对于 Pascal 语言	无	无	无

## 图的遍历（dfs）

### 【题目描述】

对于一个有向图G来说，我们存在一个经典的遍历算法，就是DFS（深度优先搜索遍历）。将G以1号点为起点进行DFS后，我们可以得到G的一棵DFS遍历树T。就此，我们可以把G中的所有边分成4种类型，如下：

- 树边：边  $(u, v)$  满足是T上的边。
- 返祖边：边  $(u, v)$  满足在T上v是u的祖先。
- 前向边：边  $(u, v)$  满足在T上u是v的祖先，但不是父亲（即不是树边）。
- 横叉边：除去以上所有情况的边就是横叉边。

现在我们给出四个整数A, B, C, D，要求构造图G，使得对G以1号点开始进行DFS后，树边个数为A，返祖边个数为B，前向边个数为C，横叉边个数为D。要求构造的G满足，联通（可以从1号点出发到达其他所有点）、无重边、无自环，否则视为不合法。你只需要输出其中任意一个解即可，我们会用spj判断你的解是否合法。

### 【输入格式】

仅一行，四个整数，表示 A, B, C, D。

### 【输出格式】

如果不存在解，输出-1。

如果有解，请现在第一行输出一个整数N，表示G的点数。

接下来N行，第i行先输出一个数Degreei，表示第i个点的出度；

接下来输出Degreei个数，描述i的所有出边。

注意不能存在重边和自环。

请注意，对于点i，在DFS过程中，我们会根据你输出的边的顺序，依次进行拓展。

### 【样例输入】

3 1 1 1

### 【样例输出】

4

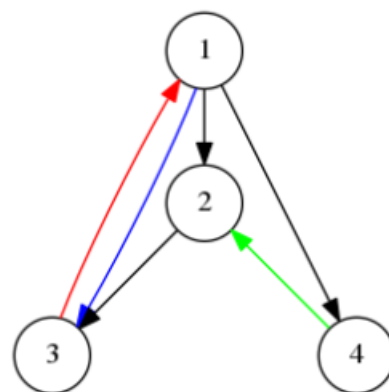
3 2 4 3

1 3

1 1

1 2

### 【样例说明】



如图所示，黑色的边为树边，红边是返祖边，蓝边是前向边，绿边是横叉边。

## 【数据范围】

本题一共 10 个测试点，每个测试点分值为 10 分。测试数据可分为 6 个部分，具体分值和说明如下：

- 10% 的数据满足： $0 \leq A, B, C, D \leq 5$ 。
- 10% 的数据满足： $D=0$
- 10% 的数据满足： $B=C=0$ 。
- 20% 的数据满足： $A \leq 100$ 。
- 20% 的数据满足： $A \leq 1000$ 。
- 30% 的数据满足： $A \leq 100000$ 。

对于 100% 的数据， 满足  $0 \leq A, B, C, D \leq 100000$ 。

## 注意

由于本题的输出量过大，建议选手使用**输出优化**来加速输出。出题人所使用的输出优化如下：

```
void Out(int x) {
    if (x < 10) {
        putchar('0' + x); return;
    }
    Out(x/10);
    putchar('0' + x%10);
}
```

## EASY (easy)

### 【题目描述】

由于出题人很懒，就没有背景了。

有  $n$  个数  $a_i$ ，现在有两个操作：

0 l val：把  $a_i$  的值换成 val。

1 l r k：选出最多  $k$ （可以不选）个  $l \leq x_1 \leq y_1 < x_2 \leq y_2 < \dots < x_k \leq y_k \leq r$ ，求  $\text{Max} \{ \sum_{j=x_i}^{y_i} a_j \mid i=1 \dots k \}$

简化题意：在  $[L..R]$  中选不超过  $k$  段不相交的子段，使这些子段内值总和最大

### 【输入格式】

第一行一个数  $n$ ，为数字个数。

第二行  $n$  个整数，为  $a_i$ 。

第三行一个数  $m$ ，为操作个数。第四行到第  $m+3$  行，每行第一个数  $op$  为操作编号，接下来若干个数对应该操作。

### 【输出格式】

若干行，对每个操作输出对应的答案。

### 【样例输入 1】

9

9 -8 9 -1 -1 -1 9 -8 9

3

1 1 9 1

1 1 9 2

1 4 6 3

**【样例输出 1】**

17

25

0

**【样例输入 2】**

15

-4 8 -3 -10 10 4 -7 -7 0 -6 3 8 -10 7 2

15

1 3 9 2

1 6 12 1

0 6 5

0 10 -7

1 4 9 1

1 7 9 1

0 10 -3

1 4 10 2

1 3 13 2

1 4 11 2

0 15 -9

0 13 -9

0 11 -10

1 5 14 2

1 6 12 1

**【样例输出 2】**

14

11

15

0

15

26

18

23

8

**【数据范围】**

15%:  $n, m \leq 50$

另 15%:  $\sum k * (r - l + 1) \leq 100000$

100%:  $n, m \leq 100000, |a_i| \leq 1000, 0 \leq k \leq 20, 1 \text{ 操作数量} \leq 10000$

## 字符串匹配 (string)

### 【题目描述】

HZY 看到了一个  $2 \times N$  的矩阵  $A$ ，矩阵中每个格子都是一个小写字母。同时，你得到了长度为  $M$  一个字符串  $S$ ，你需要在矩阵中找到一条不重复路径（起点和终点任意），使得依次经过的字母连起来恰好是  $S$ ，求这样的路径有多少种。

你只能向上、向下、向左或向右走，不能斜着走或跳着走，也不能走出矩阵外或重复经过同一个格子。两种路径不同，当且仅当至少有一个时刻所在的位置不同。

由于答案可能很大，你只要输出答案对  $1e9+7$  取模的值即可。

### 【输入格式】

第一行和第二行两行长度相同的字符串描述  $2 \times N$  的矩阵  $A$ 。

第三行一个空行。

第四行一个字符串，表示  $S$ 。

### 【输出格式】

一个正整数表示答案对  $1e9+7$  取模后的值若干行，表示操作 2 和操作 3 的答案。

### 【样例输入 1】



code

edoc

code

【样例输出 1】

4

【样例输入 2】

aaa

aaa

aa

【样例输出 1】

14

【样例说明】

对于第一个样例，字符串 S 为 code。  
矩阵 A 为：

c	o	d	e
e	d	o	c

4 种路径如下：

c	o	d	e
e	d	o	c

c	o	d	e
e	d	o	c

c	o	d	e
e	d	o	c

c	o	d	e
e	d	o	c

【数据范围】

对于测试点 1，保证  $N \leq 5$ 。  
对于测试点 2~3，保证  $N \leq 10$ 。

对于测试点 4, 保证  $N \leq 20$ 。

对于测试点 5~6, 保证  $N \leq 50$ 。

对于测试点 7~8, 保证  $N \leq 300$ 。

对于测试点 9~10, 保证  $N \leq 1,000$ 。

对于测试点 11~12, 保证  $S$  是一个全  $a$  字符串, 且  $A$  也是全  $a$  的矩阵。

对于测试点 13~14, 保证  $S$  是一个全  $a$  字符串。

对于测试点 15~16, 保证  $S$  是一个形如  $abbbb\cdots$  的仅由一个字符  $a$  和若干字符  $b$  组成

的字符串。

对于 100% 的数据, 保证  $1 \leq N, M \leq 2 \times 10^3$ 。