

压缩字符串

(a.cpp/in/out, 1S, 512MB)

【题目描述】

现在有一个长度为 N 的字符串，需要对它压缩。压缩的规则如下：

1. 一个字符串 S 可以压缩成它自身，即 $S=S$ 。
2. X 个连续的 S 可以压成 $X(S)$ ，即 $X(S)=SSS\cdots SS$ (X 个 S)。
3. 如果 $A=A'$ 、 $B=B'$ ，则 $AB=A' B'$

给定字符串，求它压缩后的最短长度。

例如字符串 AAAAAAABABABD 压缩后为 7(A)3(AB)D，所以其压缩后的最短长度为 10。

【输入格式】

仅一行，即长度为 N 的字符串 S 。

【输出格式】

仅一行，即压缩后的字符串的最短长度。

【输入样例】

TYESNONONONONONYESNONONONONONYESNONONONONONON

【输出样例】

12

【样例解释】

可以压缩成：T3(YES5(NO))，其长度为 12。

【数据范围】

- 对于另外 10% 的数据，所有的 $S[i]$ 相同；
- 对于另外 10% 的数据，所有的 $S[i]$ 互不相同；
- 对于另外 20% 的数据， $N \leq 10$ ；
- 对于 100% 的数据， $N \leq 100$

点心分配

(b.cpp/in/out, 1S, 512MB)

【题目描述】

考虑到高三住校生学习的辛苦，学校食堂计划为他们配送点心。

现在高三的学生分住在两栋楼，配送的点心有三种：面包、蛋挞和夹心饼干。学校计划配送 N 次的点心，并且每次只能配送给一栋宿舍楼。学生们喜欢每次能分到不一样的点心，所以当每次点心送到宿舍楼时，学生们就会比较这次点心和他们得到的前两次（或者少于两次，如果前面运送点心的次数不足两次）的点心。如果这几次都是同一类型的点心，则学生们学习的成绩会提高 1 分，如果这几次点心中有两种不同的类型，则学生们学习的成绩会提高 2 分，如果这几次点心中有三种不同类型的点心，则学生们学习的成绩会提高 3 分。

现在已经知道点心的类型及其被配送的顺序，要求你写一个程序，确定如何来配送，从而能使得两栋宿舍楼提高的分数总和最大。

注意，两栋宿舍楼并不要求接收相同数量的点心。事实上，也允许将所有点心都送给一栋宿舍楼。

【输入格式】

输入的第一行包含一个整数 N ，表示配送点心的次数。

第二行由 N 个字符组成的字符串，按照配送顺序依次表示点心的类型，每个字符是以下三个大写字母之一，'M'、'F' 或 'B'。

【输出格式】

输出一个整数，表示最大提高的分数总和。

【输入样例】

```
11
MBBFBFFBBMM
```

【输出样例】

```
23
```

【数据范围】

对于 30% 的数据， $N \leq 20$ ；

对于 100% 的数据， $N \leq 10^5$ 。

树

(c.cpp/in/out, 1S, 512MB)

【问题描述】

给定一个以 1 为根的 N 个结点的树，结点依次编号为 $1 \sim N$ 。

小 X 要从根结点走到叶结点，从一个结点 u 走到另外一个结点 v ，需要耗费 1 天的时间，由于小 X 的智商很低，到达一个结点后，总是随机选择一个新的可以到达的结点前进，即走到一个结点后不会再返回。

树的结点中有 K 个结点称为 P 结点，即小 X 到达 P 结点会因为各种问题不得不停留一天，然后才能继续前进。如果某个叶结点是 P 结点，那么在该结点逗留几天显然不是问题。

而小 X 目前准备的物资只够 M 天使用。现在他想知道他最不可能到达和最可能到达的叶结点的编号。

假如有多个叶结点满足，两个答案都输出编号最小的。

【输入格式】

第一行三个整数 N 、 M 、 K ，分别表示结点的数量、物资可以承受的天数和 P 结点的数量；

第二行 K 个整数（若 K 不等于 0），表示 P 结点的编号；

接下来行 $N-1$ 行，每行两个整数 u 、 v ，表示 u 、 v 之间有一条边。

【输出格式】

输出两行。

第一行表示最不可能到达的叶结点，如果不存在这样的结点，则输出 0。

第二行表示最可能到达的叶结点，如果不存在这样的结点，则输出 0。

【输入样例】

6 1 0

1 2

1 3

2 4

2 6

4 5

【输出样例】

5

3

【数据范围】

20%数据： $2 \leq N \leq 20$, $2 \leq M \leq 50$, $0 \leq K \leq 5$;

100%数据： $2 \leq N \leq 1000$, $2 \leq M \leq 1000$, $0 \leq K \leq 100$

跳板

(d.cpp/in/out, 1S, 512MB)

【题目描述】

B 在一个仅允许沿平行于坐标轴方向移动的二维方阵中。她从点 $(0, 0)$ 出发，想要到达 (N, N) ($1 \leq N \leq 10^9$)。

为了帮助她达到目的地，方阵中有 P ($1 \leq P \leq 10^5$) 个跳板。每个跳板都有其固定的位置 (x_1, y_1) ，如果 B 使用它，会落到点 (x_2, y_2) 。

B 是一个过程导向的奶牛，所以仅允许她向上或向右行走，从不向左或向下。类似地，每个跳板也设置为不向左或向下。B 需要行走的距离至少是多少？

【输入格式】

输入第一行包含两个空格分隔的整数 N 和 P 。

以下 P 行，每行包含四个整数 x_1, y_1, x_2, y_2 ，其中 $x_1 \leq x_2$ 且 $y_1 \leq y_2$ 。

所有跳板的位置和目标位置均不相同。

【输出格式】

输出一个整数，为 B 到达点 (N, N) 需要行走的最小距离。

【样例输入】

```
3 2
0 1 0 2
1 2 2 3
```

【样例输出】

```
3
```

【样例解释】

B 的最佳路线为：

- B 从 $(0, 0)$ 走到 $(0, 1)$ (1 单位距离)。
- B 跳到 $(0, 2)$ 。
- B 从 $(0, 2)$ 走到 $(1, 2)$ (1 单位距离)。
- B 跳到 $(2, 3)$ 。
- B 从 $(2, 3)$ 走到 $(3, 3)$ (1 单位距离)。

B 总共走过的路程为 3 单位距离。

【数据范围】

对于 30% 的数据，满足 $P \leq 1000$ ；

对于 100% 的数据，没有额外限制。