

CDQZ OI Test for DP

2013 OI Team

May 19, 2012

题目名称	文件名	时间限制	内存限制
子串与子序列	substr.*{in,out,cpp,pas,c}	2s	256MB
人品保卫战	defence.*{in,out,cpp,pas,c}	1s	256MB
怪盗与新一的乱步	museum.*{in,out,cpp,pas,c}	10s	256MB

Contents

1	子串与子序列	2
1.1	题目描述	2
1.2	输入格式	2
1.3	输出格式	2
1.4	样例输入	2
1.5	样例输出	2
1.6	Hint	2
2	人品保卫战	3
2.1	题目描述	3
2.2	输入格式	3
2.3	输出格式	3
2.4	样例输入	3
2.5	样例输出	3
2.6	Hint	3
3	怪盗与新一的乱步	4
3.1	题目描述	4
3.2	输入格式	4
3.3	输出格式	4
3.4	样例输入	4
3.5	样例输出	4
3.6	Hint	4

1 子串与子序列

1.1 题目描述

给你两个字符串 s 与 t , 你需要统计不同二元组 (x, y) 的个数, 满足 x 是 s 的子串, y 是 t 的子序列, 且 $x = y$ 。两个二元组 $(x_1, y_1), (x_2, y_2)$ 不同, 当且仅当 x_1 和 x_2 是 s 中不同子串或 y_1 与 y_2 是 t 中不同子序列。

关于子串与子序列的解释:

s 的一个子串 x 是一个非空的字符串, 满足 $x = s[a..b] = s_a s_{a+1} \dots s_b$, $1 \leq a \leq b \leq |s|$ 。

s 的一个子序列 x 是一个非空的字符串, 满足 $x = s[p_1 p_2 \dots p_{|y|}] = s_{p_1} s_{p_2} \dots s_{p_y}$, $1 \leq p_1 < p_2 < \dots < p_y \leq |s|$ 。

1.2 输入格式

两行, 分别表示 s 与 t 。均只包含小写英文字母。

1.3 输出格式

输出不同二元组的组数模 $1000000007(10^9 + 7)$ 。

1.4 样例输入

```
aa
aa
```

1.5 样例输出

```
5
```

1.6 Hint

50% 的数据满足 $N \leq 250$;

100% 的数据满足 $N \leq 5000$ 。

最好用 long long。

2 人品保卫战

2.1 题目描述

设想你去参加 *noip2012*, 你的座位是在第一行的末端 (即第 $N + 1$ 个)。第一行 $N + 1$ 个座位, 你必须从第一个座位走到第 $N + 1$ 个座位。初始时, 经过每个座位的时间为 T 秒。不幸的是, 前 N 个位置上都坐满了杀气冲天的神牛。这里假设神牛只有三种:

1. *TedYin* 经过 *Ted* 所在位置时每秒钟 rp 会减少 R 点。
2. *TheodoreYou* 当你走过 *Theo* 所在位置, 之后的每一秒都会因为 *Theo* 强大的气场被吸走 G 点 rp 。
3. *BeforeRain* 当你走过 *BeforeRain* 所在位置, 你会因见到珍惜的 *OImei* 而放缓前进的步伐。即经过之后每个人的时间会增加 B 秒。

其中, *Theo* 和 *BeforeRain* 的效果可以叠加。也就是说, 在经过 m 个 *Theo* 后, 每秒会被吸走 mG 点 rp 。在经过 m 个 *BeforeRain* 后, 经过每个座位的时间会变为 $T + mB$ 。

现在, 离 *noip2012* 还有一段时间。假设你已经知道你的位置, 你必须预测到最坏情况下会被吸走多少 rp 。写个程序输出这个最大值。

注意: 所有神牛的影响在进入你的位置时就消失了。

2.2 输入格式

一行, 五个空格隔开的整数 N 、 R 、 G 、 B 、 T 。

2.3 输出格式

只需输出一行一个整数, 即最坏情况下会被吸走多少 rp 。

2.4 样例输入

5 4 3 2 1

2.5 样例输出

82

2.6 Hint

20% 的数据满足 $N \leq 12$;

50% 的数据满足 $N \leq 100$;

100% 的数据满足 $N \leq 1024$, $0 \leq R, G, B \leq 65536$, $0 \leq T \leq 3$ 。

3 怪盗与新一的乱步

3.1 题目描述

怪盗基德又一次发出预告函啦!!

这一次的目标是位于 QZ 科技楼的人体标本。怪盗命运中的宿敌——工藤新一自然不会让基德得手。为了简化问题,可以将错综复杂科技楼视为一个含有 n 个点, m 条双向边的无向图。一开始,基德在 a 点,新一在 b 点。由于两人实力相当,都无法预测对手的行动,故都以以下规则移动:

在第 i 个顶点,以 p_i 的概率停留在 i 点, $1 - p_i$ 的概率离开 i 点,以等概率进入相邻的顶点(即有边与 i 相连)。

若两人在同一顶点相遇,则会进行决斗。我们都非常想知道决斗的地点,以方便围观。希望你能写个程序来计算在每个顶点决斗的概率。

3.2 输入格式

第一行包含四个整数: n, m, a, b 。以下 m 行每行两个整数 u, v 表示顶点 u 与 v 有一条边相连。最后 n 行每行一个实数 p_i ($0.01 \leq p_i \leq 0.99$)。

3.3 输出格式

输出 n 行,对应每个顶点决斗的概率。只用保留一位小数。

3.4 样例输入

```
2 1 1 2
1 2
0.5
0.5
```

3.5 样例输出

```
0.5
0.5
```

3.6 Hint

50% 的数据满足 $n \leq 5$;

100% 的数据满足 $n \leq 20, n - 1 \leq m \leq \frac{n(n-1)}{2}$ 。保证没有重边,自环。