

NOIP 2017 Round

Duration: 4h

Wearry

July 11, 2017

Problem	A	B	C
Time limit	1s	2s	1s
Memory limit	128M	128M	128M
Source file name	A	B	C
Input file suffix	.in	.in	.in
Output file suffix	.out	.out	.out
Compile options	-O2 -lm	-O2 -lm	-O2 -lm
Test cases	10	20	10
Scores pre case	10	5	10

A.cpp/in/out

Description:

YYC 正在玩一个积累 RP 的游戏, 游戏在一个长度为 N 的序列上进行, YYC 在每一个位置上会有 P_i 的概率得到一点欧气, $1 - P_i$ 的概率得到一点非气, 然后用最终得到的序列给 YYC 计算 RP 变化值, 计算方式如下:

1. 对于所有**极长连续全欧序列**, 设其长度为 x , 它会使得 YYC 的 RP 增加 $x^3 + K$
2. 对于所有**极长连续全非序列**, 设其长度为 x , 它会使得 YYC 的 RP 降低 $x^2 + K$

现在 YYC 想知道自己的 RP 在期望下会变化多少?

Input:

输入数据的第一行包含两个空格隔开的整数 N, K

接下来一行包含 N 个由空格隔开的浮点数 $P_1, P_2, P_3, \dots, P_N$, 意义如题所述.

Output:

输出仅一行一个浮点数, 表示 YYC 的 RP 的期望变化值.

注意: 当你的答案与标准答案误差不超过 10^{-3} 视为正确.

Sample Input:

```
16 1
0.642214 0.390778 0.670612 0.070373 0.717792 0.816008 0.499635 0.370708
0.330759 0.028899 0.220661 0.154958 0.700083 0.720874 0.659384 0.841728
```

Sample Output:

36.795074

Agreement:

对于 20% 的数据, $N \leq 20$.

对于 60% 的数据, $N, K \leq 10^3$.

对于另外 20% 的数据, $K = 0, N \leq 10^6$.

对于 100% 的数据, $K \leq 10^3, N \leq 10^6, 0 \leq P_i < 1$.

B.cpp/in/out

Description:

YYC 通过自己的努力, 在 N 场攒 RP 游戏中取得了令人震惊的成绩 (为了方便统计这些成绩已经化为整数), 然而 YYC 发现自己的成绩比 ZJW 差很多, 他很不高兴, 可惜无法在短时间内提高自己的得分.

不甘心的 YYC 想到了作弊, YYC 将会对这个成绩序列进行 M 次篡改, 每次篡改他会随机得到一个区间 (这样的区间共有 $N \times (N + 1)/2$ 个), 然后将这个区间内的比赛成绩全部改为区间内比赛成绩的最大值.

ZJW 想知道 YYC 能不能超过自己, 于是请你算出 YYC 每场比赛的期望最终成绩.

自信的 YYC 认为自己的成绩一定很好, 会有很大的浮点误差, 于是他钦定你将答案对 998244353 取模. 若答案能够表示成为一个分数 p/q , 那么输出 $p \times q^{-1} \bmod 998244353$, 这里 q^{-1} 表示乘法逆元.

Input:

输入数据的第一行包含由空格分隔的两个正整数 N, M .

接下来一行包含 N 个由空格隔开的整数 $A_1, A_2, A_3, \dots, A_N$, 表示 N 场比赛中 YYC 的成绩.

Output:

输出 N 行, 每行一个整数, 第 i 行的数表示第 i 场比赛中 YYC 的期望最终得分.

Sample Input:

```
5 4
2 1 8 9 7
```

Sample Output:

474760091
828863245
402492132
9
602279031

Agreement:

对于 20% 的数据, $N, M \leq 5$

对于 50% 的数据, $N \leq 50$

对于 100% 的数据, $1 \leq N, M \leq 500$, $A_i \leq 10^9$, 保证所有 A_i 随机生成.

C.cpp/in/out

Description:

YYC 发现自己的人品实在是太差了, 以至于通过作弊都无法比 ZJW 的分数高, YYC 感到生无可恋, 于是找到了一棵树....

这棵树共有 N 个节点, 其中 1 号节点为根. 树上每一个节点有一颗果实, 每一颗果实有一定的美味度 V_i , 同时有一定的毒素 P_i ,

YYC 要想摘下树上的某一颗果实当且仅当这个果实的父亲节点被选择, 当摘下某一颗果实时, YYC 会获得这颗果实的美味度, 同时也会中这颗果实的毒. YYC 的最多能接受的毒为 M , 现在 YYC 希望在不中毒太深的情况下获得最大的美味度.

YYC 自信地选择了一种策略, 从根节点出发每次随机摘一颗可行的果实, 但是 YYC 现在对自己的 RP 也不够自信了, 所以他希望你能帮帮他.

Input:

输入数据的第一行包含两个由空格隔开的整数 N, M .

接下来 N 行, 每行两个空格隔开的整数, V_i, P_i , 表示 i 号果实的美味度和毒素.

再接下来 $N - 1$ 行, 每行两个 $[1, N]$ 之间的整数, u, v , 表示 u, v 之间由连边.

Output:

输出仅一行一个整数, 表示 YYC 在不超过毒素上限的情况下能够获得的最大美味度.

Sample Input:

```
7 9
39 6
13 2
```

22 6
7 4
-19 5
28 6
-17 1
2 1
3 2
4 1
5 4
6 2
7 3

Sample Output:

52

Agreement:

对于 20% 的数据, 满足树的形态是一条链.

对于 50% 的数据, 满足 $N, M \leq 500$.

对于 100% 的数据, 满足 $N, M \leq 2000, 1 \leq |V_i| \leq 10^4, 1 \leq P_i \leq 10^4$