

CSP2019 模拟试题

第二试

(请选手务必仔细阅读此页内容)

一. 题目概况

中文题目名称	贪玩蓝月	超级加倍	校庆
英文题目名称	moon	super	celebration
可执行文件名	moon	super	celebration
输入文件名	moon.in	super.in	celebration.in
输出文件名	moon.out	super.out	celebration.out
每个测试点时限	1 秒	1 秒	3 秒
内存上限	512MB	512MB	1024MB
测试点数目	20	20	20
每个测试点分值	5	5	5
结果比较方式	全文比较	全文比较	全文比较
题目类型	传统	传统	传统
是否开启 O2	是	是	是

二. 提交源程序文件名

对于 Pascal 语言	moon.pas	super.pas	celebration.pas
对于 C 语言	moon.c	super.c	celebration.c
对于 C++ 语言	moon.cpp	super.cpp	celebration.cpp

三. 编译命令

对于 Pascal 语言	fpc moon.pas	fpc super.pas	fpc celebration
对于 C 语言	gcc -o moon moon.c -lm	gcc -o super super.c -lm	gcc-o celebration celebration.c -lm
对于 C++ 语言	g++-O2 -o moon moon.cpp -lm	g++ -O2 -o super super.cpp -lm	g++ -O2 -o celebration celebration.cpp -lm

贪玩蓝月 (moon)

【题目描述】

在贪玩蓝月这款游戏中，一共有 m 个副本，其中打每一个副本需要 w_i 的代价。

机房一共有 n 个小伙伴，每一个小伙伴颓废的时间为 t_i 。对于每一个小伙伴，他会顺次遍历所有副本，如果颓废的时间 \geq 打当前副本的时间，他会选择打副本并消耗相应的时间；否则他会遍历下一个副本。

现在，小伙伴们想知道自己能打多少个副本。你能帮帮他吗？

【输入数据】

输入第一行包含两个数， m 、 n 。分别表示副本数和小伙伴的数量。

接下来输入 m 个数 w_i ，每一个数表示打第 i 个副本需要的时间。

接下来输入 n 个数 t_i ，每一个数表示第 i 个小伙伴颓废的时间。

【输出数据】

输出数据共 n 行，第 i 行输出的数为第 i 个小伙伴能够打的副本数。

【输入样例】

```
5 5
3 2 4 1 5
7 4 3 100 9
```

【输出样例】

3
2
1
5
3

【样例解释】

对于第一个小伙伴，他打的副本为 1、2、4；

对于第二个小伙伴，他打的副本为 1、4；

对于第三个小伙伴，他打的副本为 4；

对于第四个小伙伴，他打的副本为 1、2、3、4、5；

对于第五个小伙伴，他打的副本为 1、2、3。

【数据约定】

对于 20%的数据，有 $n, m \leq 1000$ 。

对于 40%的数据，有 $n, m \leq 5000$ 。

另有 20%的数据， w_i 满足单调不降

对于 100%的数据，满足 $1 \leq n, m \leq 1000000, 1 \leq w_i, t_i \leq 1000000$

超级加倍(super)

【题目描述】

在欢乐斗地主中，一共有 n 种加倍卡，其中每一种都能使倍数变成原来的 w_i 倍。特别的，在此我们认为底倍数为 1。

当然，仅仅一次加倍是不能满足人们心中的快感的，所以腾讯推出了新的加倍方式——「超级加倍」。

定义一次「超级加倍」为进行 m 次加倍操作，其中每次加倍操作可以在 n 种加倍卡中任意选择一张卡进行加倍。特别的，在「超级加倍」中，每一种加倍卡的出现次数不限。

作为策划师的你，为了判断「超级加倍」的强度，你需要回答所有本质不同的「超级加倍」的倍数的和。当然，由于欢乐豆是有限的，你只需要回答对 998244353 取模后的结果。

「超级加倍」A、B 本质不同定义为：存在至少一个 i ，使得第 i 种加倍卡在 A 中的出现次数不等于在 B 中的出现次数。

例如，如果 $m=2$ ， $n=2$ ，2 种加倍卡倍率分别为 w_1 和 w_2 ，则共有 4 种情况：

1. 第一次选 w_1 ，第二次选 w_2 ，总倍率为 w_1w_2 。
2. 第一次选 w_2 ，第二次选 w_1 ，总倍率为 w_1w_2 。
3. 第一次选 w_1 ，第二次选 w_1 ，总倍率为 w_1w_1 。
4. 第一次选 w_2 ，第二次选 w_2 ，总倍率为 w_2w_2 。

在第一种和第二种方案中，都获取了一张 w_1 和一张 w_2 ，所以我们

认为它们是本质相同的，所以上面例子中共有三种本质不同的方案，权值和为 $(w_1w_2+w_1w_1+w_2w_2)$ 。

注意，即便存在 i 、 j 的加倍倍数一样， i 、 j 仍视作**本质不同**的两种加倍卡；比如在上一个例子中，如果 $w_1=w_2$ ，本质不同的超级加倍的方案仍然有三种，而不是一种。

【输入数据】

第一行两个数，分别为 n 、 m 。

第二行 n 个数，第 i 个表示 w_i 。

【输出数据】

一行一个整数，表示所有**本质不同**的「超级加倍」的倍数的和。

【输入样例】

```
2 3
2 3
```

【输出样例】

```
65
```

【样例解释】

一共有四种本质不同的超级加倍：

选择 0 张第 1 种卡，3 张第 2 种卡，此时倍数为 $3*3*3=27$ ；

选择 1 张第 1 种卡，2 张第 2 种卡，此时倍数为 $2*3*3=18$ ；

选择 2 张第 1 种卡，1 张第 2 种卡，此时倍数为 $2*2*3=12$ ；

选择 3 张第 1 种卡，0 张第 2 种卡，此时倍数为 $2*2*2=8$ ；

所以倍数和为 65

【数据约定】

对于 10%的数据，有 $n, m \leq 7$ 。

对于 30%的数据，有 $n, m \leq 100$ 。

对于 50%的数据，有 $n \leq 5, m \leq 10^5$ 。

对于 70%的数据，有 $n \leq 200, m \leq 10^5$ 。

对于 100%的数据，有 $n \leq 200, m \leq 10^9, 0 \leq w_i < 998244353$

校庆(celebration)

【题目描述】

风景怡人的华懿高级中学一共有 n 个景点，有 $n-1$ 条道路将其连接。游客参观第 i 个景点时会收获 w_i 愉悦度。但是如果多次参观类似的景点会令人心生厌烦，所以总共获得的愉悦度为：所有选择的景点的愉悦度的异或和。

由于正值华懿高级中学 70 周年校庆，所以每一天都会翻新一条从 x 到 y 的路径上的所有景点，此时这些景点将不能被访问。为了安排参观，现在主办方想知道：在剩下的所有景点中，有多少种选择方案能够使得总愉悦度为 k 。

为了避免给你带来太多困扰，主办方只需要对 998244353 取模后的结果。你能够帮帮他吗？

【输入数据】

第一行两个整数 n, m ，表示景点数和询问数。

接下来 $n-1$ 行，每一行两个数 x, y ，表示原图的一条边 (x, y) 。

接下来一行 n 个数，第 i 个数表示 i 号景点的愉悦度为 w_i 。

接下来 m 行，每行三个数 x, y, k ，表示原图的 (x, y) 这条路径正在整修，主办方想知道有多少种选择方法使得异或值为 k 。

【输出数据】

一共输出 m 行，第 i 行表示第 i 组询问的答案。

【输入样例】

5 3
1 2
1 3
2 4
2 5
1 6 3 4 2
2 3 2
1 1 4
4 4 2

【输出样例】

1
2
2

【输出样例】

对于第一天，删去 2→3 这条路径上的景点后，剩下景点为 4、5；
其中选择（5）可以使得愉悦度为 2.

对于第二天，删去 1→1 这条路径上的景点后，剩下景点为 2、3、
4、5；其中选择（4）、（2，5）可以使得愉悦度为 2.

对于第三天，删去 4→4 这条路径上的景点后，剩下景点为 1、2、
3、5；其中选择（5）、（1，3）可以使得愉悦度为 2.

【数据约定】

对于 10%的数据，有 $n, m \leq 5$ 。

对于 30%的数据，有 $n, m \leq 1000$ 。

对于 60%的数据，有 $n, m \leq 50000$ 。

另有 20%的数据，保证询问时 x 为 y 的祖先。

对于 100%的数据， $1 \leq n, m \leq 200000$, $0 \leq w_i, k \leq 32767$.