信息学联赛模拟试题

题目名称	不同位置	唉舞萌痴	代码调试	向食堂奔去
题目类型	传统型	传统型	传统型	传统型
每个测试点时限	1.0 秒	1.0 秒	1.0 秒	2.0 秒
内存限制	256 MiB	256 MiB	256 MiB	512 MiB
测试点数目	10	20	10	20
编译选项	对于 C++ 语言 `-O2 -std=c++14 -static`			

注意事项（请仔细阅读）

1. 文件名（程序名和输入输出文件名）必须使用英文小写。
2. C/C++ 中函数 `main()` 的返回值类型必须是 `int`，程序正常结束时的返回值必须是 0。
3. 提交的程序代码文件的放置位置请参考考试平台的具体要求。
4. 因违反以上三点而出现的错误或问题，一律不予受理。
5. 若无特殊说明，结果的比较方式为全文比较（过滤行末空格及文末回车）。

不同位置（string）

【题目背景】

一零圈曾经出过一个题叫 Popcount Xor（当然本题和这个题并没有什么关系），就在 loj 上。原做法是一个不太好看的东西，不过被更常规的分治爆掉了 std。后来 zak 又提出了线性做法，不过我并看不懂就是了（哭

其实那个题算是为数不多的独立出的个人觉得还可以的题，嗯，所以只是打个广告（）

【题目描述】

给定一个长为 n 的 01 串 s。**保证 s 第一位是 1**。再给定一个 k。求字典序最大的长为 n 的满足下列条件的 01 串 t:

1. s 的字典序大于等于 t。 2. s 和 t 的不同位的个数必须大于等于 k。

定义长为 n 的串 a 字典序小于串 b，当且仅当 a 和 b 有不同位，且第一个不同位 i 满足 $a_i = 0, b_i = 1$。

【输入格式】

第一行一个串 s。

第二行一个非负整数 k。

【输出格式】

一行一个字符串表示答案。如果无解则输出 -1。

【样例】

【样例输入 1】

```
1100000
4
```

【样例输出 1】

```
1011111
```

【样例输入/输出 2】

见下发文件 string2.in/ans。该样例满足测试点 3 ~ 4 的限制。

【样例输入/输出 3】

见下发文件 string3.in/ans。该样例满足测试点 9～10 的限制。

【数据范围与提示】

本题共 10 个测试点，每个测试点 10 分。

测试点编号	特殊性质
1	k = 0
2	k = n
3～4	s 的倒数第 k 位是 1
5～7	n ⩽ 10
8	n ⩽ 1000
9～10	无特殊限制

对于所有数据，$1 \leqslant n \leqslant 10^6$，$0 \leqslant k \leqslant n$。

唉舞萌痴（altra）

【题目背景】

其实小铃是一个舞萌痴。

【题目描述】

小铃所在的国家有 n 个城市。这些城市之间由 n－1 条道路连接，并且任意两个城市之间可以经由若干道路互达（也就是说这些城市形如一棵树）。

出于一些诡异的攀比心（？），小铃想去每一个城市打卡舞萌。

小铃所在的城市是 1 号城市，小铃天天玩这座城市的舞萌，所以不需要额外打卡。

为了节约时间，小铃决定多次从 1 号城市出发，打卡所有城市。对于某次出行，小铃会选择一个没有去过的，只有一条道路连接的城市为终点，沿若干条道路行进到终点后立刻原路返回。期间所有没到达过的城市小铃都会去打卡舞萌。

出于想考考大家的心情（？），小铃想知道当她打卡了所有城市的舞萌后，这每一次的出行中，打卡舞萌个数的最大值减去打卡舞萌个数的最小值（也就是极差）最小是多少。

【输入格式】

第一行一个正整数 n。

第二行 n－1 个正整数，第 i 个正整数表示 i + 1 号城市和它有道路相连，保证这个数 ≤ i 。

【输出格式】

一行一个整数表示答案。

【样例】

【样例输入 1】

```
6
1 2 2 1 1
```

【样例输出 1】

```
1
```

【样例 1 解释】

小铃依次选择了 5,6,3,4 号城市，分别打卡了 1,1,2,1 个城市的舞萌。

【样例输入/输出 2】

见下发文件 maimai2.in/ans。该样例满足测试点 7 ~ 8 的限制。

【样例输入/输出 3】

见下发文件 maimai3.in/ans。该样例满足测试点 9 ~ 10 的限制。

【数据范围与提示】

本题共 20 个测试点，每个测试点 5 分。

测试点编号	特殊性质
1 ~ 3	$n \leqslant 10$
4 ~ 6	只有一条道路相连的城市个数小于 10
7 ~ 8	只有城市 1 相连的道路数可能大于 2
9 ~ 10	$n \leqslant 100$
11 ~ 13	$n \leqslant 400$
14 ~ 16	$n \leqslant 3000$
17 ~ 20	无特殊限制

对于所有数据，$1 \leqslant n \leqslant 2 \times 10^5$。

代码调试（bug）

【题目描述】

有一份 n 行的代码。其中 m 行代码有 bug，分别位于 $a_1, a_2, \cdots, a_m$ 行，调代码的人知道所有 bug 的位置。

还需要进行一些运行的尝试来了解错误的具体细节并修复它们。

每次运行时，可以任意选一个位置 i，随后让程序从第 1 行开始一直运行到第 i 行后结束。运行这 i 行代码总共需要 i 秒。

接下来可以一次性修复所有这 i 行中的所有 bug。设这 i 行代码中现存的 bug 数量是 x，那么需要 x^4 秒来完成修复。修复后，这 i 行代码中将不再存在任何 bug。

最短需要多少秒才能修复所有 bug?

【输入格式】

第一行两个正整数 n, m。

接下来 m 个正整数表示 $a_1, a_2, \cdots, a_m$。保证 $1 \leqslant a_1 < a_2 < \cdots < a_m \leqslant n$。

【输出格式】

一行一个整数表示答案。

【样例】

【样例输入 1】

```
9 2
8 9
```

【样例输出 1】

```
19
```

【样例 1 解释】

【样例输入/输出 2】

见下发文件 bug2.in/ans。该样例满足测试点 3 ~ 4 的限制。

【样例输入/输出 3】

见下发文件 bug3.in/ans。该样例满足测试点 8 ~ 10 的限制。

【数据范围与提示】

本题共 10 个测试点，每个测试点 10 分。

测试点编号	特殊性质
1 ~ 2	$n \leqslant 15$
3 ~ 4	$m \leqslant 15$
5 ~ 6	$m \leqslant 2000$
7	$n = m$
8 ~ 10	无特殊限制

对于所有数据，$1 \leqslant m \leqslant n \leqslant 2 \times 10^5$。

向食堂奔去（canteen）

【题目描述】

即将到达饭点，生活在 T 大的小圈决定迅速赶到食堂！为了更快的到达（并且因为小圈没太大力气骑自行车），小圈可以先找到一个有共享电动车的地方，骑上电动车后迅速赶往食堂。骑上电动车的时间忽略不计。

T 大的结构可以看成一棵 n 个点的树，由 n－1 条道路相连。其中共有 m 个点有电动车，分别在 $a_1, a_2, \cdots, a_m$。在没有电动车时，经过一条道路需要 2 单位时间；有电动车时，经过一条道路只需要 1 单位时间。

现在有 q 天，每天小圈都要赶往食堂。第 i 天小圈初始时在 x_i，而食堂在 y_i。小圈想知道到达食堂的最短时间是多少。

【输入格式】

第一行两个正整数 n, m。

接下来 n－1 行，每行两个正整数表示一条道路的两端。保证其组成一棵树。

接下来一行 m 个正整数，表示有共享电动车的点的编号。

接下来一行一个正整数 q 表示询问个数。

接下来 q 行，第 i 行两个正整数表示 x_i, y_i。

【输出格式】

q 行，每行一个非负整数表示答案。

【样例】

【样例输入 1】

```
5 1
4 2
5 1
1 4
3 5
5
3
3 5
4 4
2 4
```

【样例输出 1】

```
2
0
2
```

【样例 1 解释】

【样例输入/输出 2】

见下发文件 canteen2.in/ans。该样例满足测试点 1 ~ 3 的限制。

【样例输入/输出 3】

见下发文件 canteen3.in/ans。该样例满足测试点 11 ~ 13 的限制。

【数据范围与提示】

本题共 20 个测试点，每个测试点 5 分。

测试点编号	特殊性质
1 ~ 3	$n,q\leqslant 100$
4 ~ 6	$n,q\leqslant 2000$
7 ~ 8	$m\leqslant 10$
9 ~ 10	保证树是一条链
11 ~ 13	保证树的生成方式是对于每个 $2\leqslant i\leqslant n$，连接 i 和一个在 $[1,i-1]$ 中随机选取的数后随机打乱编号
14 ~ 15	保证生成时每个点有电动车的概率是 $\frac{1}{2}$
16 ~ 17	$n,q\leqslant 10^5$
18 ~ 20	无特殊限制

对于所有数据，$1\leqslant n,q\leqslant 5\times 10^5$，$1\leqslant m\leqslant n$，a 数组互不相同。