第一题

关键字：链，模拟

【样例 1 解释】
第 1 轮：6 楼的火被扑灭，5 楼火势 1+5=6
第 2 轮：庆祝，5 楼火势 6+5=11
第 3 轮：5 楼火势 11-6+5=10
第 4 轮：5 楼火势 10-6+5=9
第 5 轮：5 楼火势 9-6+5=8
第 6 轮：5 楼火势 8-6+5=7
第 7 轮：5 楼火势 7-6+5=6
第 8 轮：5 楼火势 6-6=0，5 楼的火被扑灭
第 9 轮：庆祝
第 10 轮：4 楼的火被扑灭
第 11 轮：庆祝
第 15 轮：3 楼的火被扑灭
第 16 轮：庆祝
第 18 轮：2 楼的火被扑灭
第 19 轮：庆祝
第 22 轮：1 楼的火被扑灭
没有了，一旦第 1 楼的火被扑灭行动就立即结束，不会再倾注了。

其他的就不解释了。
另外，题目的第 5 条和第 10 条后面都有白字提示。

第二题

关键字：构图，最短路

本题只要照着题目要求构图然后用类似于最短路的方法求出球传到某个进攻球员的最大的概率 `f[i]`即可。

这个最短路用 `SPFA` 即可。

第三题

关键字：图论，归纳

首先无解是很好判断的，只要用广搜或深搜判断每个格子能不能连到边界即可。

然后可以发现一个给定的图纸，只要它有解，随便我们怎么剪，最后的结果都是一样的。那么，我们先将能用的边连好，计算每个格点的度数，可以猜想最后的答案就是跟这些度数有关的一个函数。

事实上，我们可以推出这样一个结论，最后的答案就是

    Ans = 奇度数的点的个数/2+度数为 4 的点+度数为 2 且在边界上的点

这个结论感性地想一想，还是很靠谱的。因为最后你肯定是每次选当前在边界上的两个奇度数的点连接起来，而除了边界上的那两个点，其余的点都减了偶数度数，所以本不在边界上的奇度数的点最后也总会出现在边界上。度数为 2 且在边界上的点相当于两个度数为 1 的点。度数为 4 点一旦被“碰”到了一次，就成了在边界上且度数为 2 的点。

所以上述结论是成立的。统计一下输出即可。

第四题

关键字：基环外向树，动态规划，单调队列 or DFS+线段树

每个点的出度为 1，这个图实际上就是一个基环外向森林。

题目实际上是要求在原图上的一条路径，使得这条路径上的最长上升子序列最长。

因为不同的联通块之间没有影响，所以我们只要考虑一个基环外向树的情况。因为一个点可以经过多次，所以环上的点的位置相当于可以任意调换，所以我们可以看成将这个环上的点按升序排好了。

接下来就只要考虑一棵树的情况，这里有两种方法，都是 O(NlogN)的。

第一种是 LD 想的：
我们就用类似于用单调队列求一个序列的最长上升子序列的方法来维护一个树上的单调队列。
由于从下往上 DP 不便于处理，我们不妨改成求从根往下的最长下降子序列。
这样，每次向下走时，相当于序列往后延伸了一位，用单调队列的方法对应修改即可。与普通的链不同的是，这里还要回溯，也就是要在单调队列中删除元素。这个实际上不必真正去“删元素”，注意到每往后延伸一位，单调队列最多改动一个元素，我们实际上只要将这个改动记录下来，回溯时改回来即可。

第二种是 LY 想的：
这个题目说白了就是一个二维限制的 DP，
f[i]=max{f[j]+1,a[i]>a[j],j 在 i 子树中}
我们将所有元素按 a 排序，用线段树在 DFS 序上维护已经求出来的 f 值，每次同样用线段树得到答案即可。
需要注意的是，这里是求上升序列，也即是不取等号，所以在排序的时候要以深度为第二关键字以保证不取到等号。

Std 里的是方法一，Check 里的是方法二。