A

有结论：对于一个树上的连通块，记其权值和为 S，如果其奇偶性与 k 相同且 $S \geq k$，那么这个连通块一定可以通过从叶子依次删除一些节点后可以是其权值和恰好为 k。

对于只有一个点的块，显然成立。

对于超过一个点的块，其至少有两个叶子。如果存在权值为 $0,2$ 的叶子，直接删除，不存在的话至少有两个权值为 1 的叶子，删去这两个叶子即可。

记录 $f_{u,0/1}$ 代表以 u 为根的子树中权值和为偶数/奇数的连通块权值和的最大值，可以通过简单树形DP进行转移，枚举 u 的每个儿子 v 并依次转移，初始 u 为根的子树中只加入了 u 这个结点，根据 u 的权值来得到初始的 f_u，接下来对于每一个 v，有

$$
\begin{aligned}
f_{u,0} &= \max\{f_{u,1} + f_{v,1}, f_{u,0} + f_{v,0}\} \\
f_{u,1} &= \max\{f_{u,0} + f_{v,1}, f_{u,1} + f_{v,0}\}
\end{aligned}
$$

在自底向上 DP 的过程中，如果发现某个时刻以一个结点为根的子树中存在可以在删去一些结点后权值和恰好为 k 的情况，显然可以直接贪心删，体现在 DP 过程中就是发现当 $f_{u,k\%2} \geq k$ 后将答案+1并将 $f_{u,0/1}$ 置零。

时间复杂度 $O(n)$。

B

每次操作相当于在原树上创建某个点的复制并把原来的边一同复制。

如果一条边的两个端点都没有被复制过，那么这条边一定要保留，否则删去后图一定不连通。

考虑所有被复制过的点构成的连通块以及这个块周围的一圈点，计算每个块中的删边方式再全部相乘就是最终的答案。

对于每个被复制过的点 u，考虑其已经确定好子节点 v 的子树内的删边方式，根据题意 u 子树里的结点在最后也必须连通，所以 v 的子树只有两种情况，分别对应：

- v 被复制过，但是结点 $v, v+n$ 之间不连通，并且使 $(v, v+n)$ 连通之后就可以使 v 的子树连通。
- v 没有被复制过或者 v 的子树是连通的并且子树之内不能成环。

我们把上面的两种情况分别称为 v 下面的子树形成了两块和一块，令 f_u 表示 u 下面的子树是一块的情况，g_u 表示 u 下面的子树是两块的情况，通过分析，可以得出如下限制：

- u 的所有子节点形成的子树之间不可能出现连边
- 如果同时删去 $(u, v), (u+n, v/v+n)$，那么图一定会不连通
- 如果 u 下面的以子节点 v 为根的子树是两块，那么边 $(u, v), (u+n, v+n)$ 都不能删，否则图不可能连通。
- 如果 u 下面的子节点都是一块，那么对于每个 v，如果都在 $(u, v), (u+n, v/v+n)$ 中任选一条删去相当于将每一个 v 所在的连通块分配给 u 或者 $u+n$，这样会让 u 变成两块。
- 那么如果有两个子节点 v_1, v_2 都是一块的情况，那么连接它们的两对边 $(u, v_1), (u+n, v_1/v_1+n)$ 和 $(u, v_2), (u+n, v_2+n)$ 不可能同时保留，否则 $(u, v_1, u+n, v_2)$ 将成环。

考虑计算 f_u，根据上面的限制条件，如果要使 u 变成一块且内部不能成环，那么枚举 u 的哪个子节点 v 保留 (u, v) 的一对边，它的选取方式就是 f_v 种，剩下的子节点要么是"一块+一对边删去一条"，要么是"两块"，选取方式就是 $2f_v + g_v$ 种。

于是

$$f_u=\sum_{v\in son_u}\left[f_v\prod_{v'\neq v}(2f_{v'}+g_{v'})\right]$$

记录一个 $2f+g$ 的前缀积和后缀积就可以做到 $O(1)$ 计算 $\prod$ 的结果。

再考虑计算 g_u，根据上面的限制条件，对于一块的子节点 v，那么连接 u,v 的一对边必须断掉一条，否则 u 和 $u+n$ 就会连通。所以

$$g_u=\prod_{v\in son_u}(2f_v+g_v)$$

将每个连通块的根的 f 值相乘就是答案，每次更新节点信息的时候重新做一遍就行。

时间复杂度 $O(n^2)$

C

显然符合条件的最终情况都是这条数轴被划分成了多个区间，并且每个区间的线段颜色都是相同的。

将所有线段按照右端点排序，记 $f_{i,c}$ 代表考虑前 i 条线段并且强制规定选择第 i 条线段，这条线段颜色为 c 的方案数，如果线段 i 颜色不为 c，那么方案数为 0。

考虑当前线段所在颜色相同区间的长度，令 $g(l,c)$ 代表区间 $[l,r_i]$ 内颜色为 c 的线段数量，那么有转移方程式

$$f_{i,c}=\sum_{j|r_j<l_i}f_{j,1-c}\times 2^{g(r_j+1,c)-1}$$

这样可以做到 $O(n^2)$ 进行计算。

观察到在加入线段 i 后所有满足 $x\le l_i$ 的 $g(x,c)$ 都会增加 1，相当于对一个 $f_{j,1-c}$ 的一个前缀进行了区间乘 2，然后再对一个满足 $r_j<l_i$ 的 $f_{j,1-c}$ 进行区间求和。

于是维护 $f_{i,0/1}$ 的两棵线段树，实现区间乘与单点加的操作即可。

时间复杂度 $O(n\log n)$。

D

考虑 $0\le a_i\le m$ 的限制条件，可以通过数位 DP 来进行实现。

设 $f_{i,lim,s}$ 表示枚举到第 i 个二进制位，有 lim 个数仍然被 $[0,m]$ 的限制条件限制了后续二进制位的选取，且当前二进制位的总和距离题目要求的权值还剩下 s 的情况数。

对于第 i 位，如果 m 的第 i 位是 0，那么被限制的数这一位只能够取 0，反之这些数可以通过取 0 来摆脱限制或者取 1 来继续接受限制。没有被限制的数可以随便取。

枚举取 1 的数的数量，在 m 的第 i 位是 0 的时候有转移方程式

$$f_{i,lim,s}=\sum_{j=0}^{k-lim}\binom{k-lim}{j}f_{i-1,lim,s-2^i\times(k-j)\times j}$$

在 m 的这一位是 1 的时候有转移方程式

$$f_{i,lim,s}=\sum_{j=0}^{k-lim}\sum_{l=0}^{lim}\binom{k-lim}{j}\binom{lim}{l}f_{i-1,lim-l,s-2^i\times(j+l)\times(k-j-l)}$$

合法的 s 很少，直接用 unordered_map 实现记忆化搜索即可。

E

注意到每次最后的结果一定是树上的某个连通块被感染。

于是枚举树上的每个连通块，最终局面是这个连通块被感染的概率就是其中选一个点作为初始点的概率乘上其余的点被传染的概率再乘上连通块周围的一圈点不被传染的概率。

考虑初始点的选取，记 $f_{i,j}$ 表示以 i 为根的子树被感染了 j 个点且没有选定初始点的概率， $g_{i,j}$ 表示选定了初始点的概率，规定 i 必须被感染。

假设要合并 u 和其儿子 v 的信息，转移方程式为

$$f'_{u,i} = f_{u,i} \times (1 - p_v) + \sum_{j+k=i} f_{u,j} \times f_{v,k}$$

$$g'_{u,i} = g_{u,i} \times (1 - p_v) + \sum_{j+k=i} (f_{u,j} \times g_{v,k} + g_{u,j} \times f_{v,k})$$

最终恰好 i 个点连通的概率为

$$\sum_{i=1}^{n} g_{u,i} \times (1 - p_{fa})$$

利用树上背包的技巧可以做到 $O(n^2)$。

- 树上背包：记录每个点的子树大小，for 循环枚举上界记为这个点当前的子树大小。