$T1\ Game$

算法 1： $30pts$

$n, m \le 50$

把第二个班的人看成负容量，只要求最终容量为0的最大权值，简单的01背包。

算法 2： $40pts$

$|v_i| \le 10000$

防止没开$long\ long$的小倒霉蛋爆掉，如果有根据这个性质得来的解法一定请讲讲XD。

算法 3： $100pts$

原题范围

大水题，沿用算法1的做法，中途舍弃过大的状态（因为在随机情况下，中途重新偏回来恰好为0的概率很小），为了强行制造随机情况，防止被精心构造的数据$hack$掉，$random_shuffle$一下即可。

$T2\ Meeting$

算法 1： $20pts$

$n \le 20$

2^n枚举所有状态，然后枚举每个点作为集合中心计算答案即可，复杂度$O(n \cdot 2^n)$

算法 2： $30pts$

$n \le 25$

注意到合法状态没有那么多，只有$C(n, m)$而已，依然枚举所有状态，求出答案（可能需要优秀的常数），复杂度$O(n \cdot C(n, m))$

算法 3： $10pts$

$m = 2$

只有两个点，最优集合点必然是在两点之间的路径上，也就是所有点对之间的距离和，很容易求出来，复杂度$O(n)$

算法 4： $20pts$

m是奇数，且树是一条链

由于m是奇数，那么最优答案一定在m个点最中心的点上，枚举这个点，求出以其作为中心点的答案即可，复杂度$O(n)$。

算法5：$100pts$

原题数据范围

按点考虑并不好做，我们按边考虑。对于一条边，如果在其两侧分别有x和$m-x$个关键点（不妨设$x \le m-x$），那么肯定是将集合点设置在$m-x$那个方向，这样这条边只会被x个点经过。

不妨设某条边两侧分别有s和$n-s$个点，那么我们要求的就是$\sum_{i=1}^{m-1} C(s,i)\cdot C(n-s,m-i)\cdot \min(i,m-i)$。这东西有min，不太好求，我们直接把他拆成两个类似的东西，$2\times\sum_{i=1}^{\frac{m-1}{2}} C(s,i)\cdot C(n-s,m-i)\cdot i+[m\%2==0]C(s,\frac{m}{2})\cdot C(n-s,\frac{m}{2})\cdot\frac{m}{2}$。后面那个可以单独特判暴算，我们只需要计算前面那个。

不妨设$\frac{m-1}{2}=k$，令
$ans=\sum_{i=1}^{k} C(s,i)\cdot C(n-s,m-i)\cdot i=s\cdot\sum_{i=1}^{k} C(s-1,i-1)\cdot C(n-s,m-i)$

令$G(s)=\sum_{i=1}^{k} C(s-1,i-1)\cdot C(n-s,m-i)$，则$ans=s\cdot G(s)$

我们现在考虑$G(s)$的组合意义，可以理解为$n-1$个物品里，一共要选$m-1$个，前面$s-1$个里最多选$k-1$个的方案数，要从$G(i)$变成$G(i+1)$，发现答案变少的部分就是前面$i-1$个选了$k-1$个，而i也被选中了，只有这种情况会被$G(i)$计算而不会被$G(i+1)$计算，$O(n)$递推即可计算，最后枚举每一条边算贡献再累加即可。

$T3\ Unreal$

算法1： $10pts$

$n=2,s_i=0$

怎么暴力都可以。

算法2： $40pts$

$n\le 200,s_i=0$

由于n比较小，而且$s_i=0$，所以猜测可能可以枚举一些时刻暴力计算答案。但是枚举的精度显然应该要比秒还要小很多，暂时没有想到比较能够接受的算法，希望有会的同学可以分享一下。

算法3： $20pts$

$h_i=h_j,m_i<30$

由于小时数相同，并且分针全都在表盘的前半圈，那么很显然时针的差异无论如何不会超过分针的差异，那么只需要最小化分针的差异即可。容易发现，最优情况一定是选择最小分针角度和最大分针角度的等分点。

算法4： $100pts$

原题数据范围

要求最小化最大误差，很容易想到二分答案。那么我们二分答案后，只需要判定是否存在某个时间满足误差小于二分的答案即可。不妨设二分的答案为x，对于每个指针，向前x°一直到向后x°的区间内都是合法的。

于是我们可以对时针和分针分别求出n个钟表的合法区间交集，若某种指针为空集，则肯定无解；否则，我们就可以得到时针和分针各自的范围。

如果时针的范围超过$1h$，那么任意一个分针都是合法的，此时必定有解；如果时针的范围不到$1h$，那么可以用时针范围直接计算出可被允许的分针范围，再与之前求出的分针范围求交集，若有交则有解，否则无解。

$T4\ Tree$

算法 1： $10pts$

$n \le 5$

大暴力，怎么写都可以。

算法 2： $20pts$

$n \le 7$

有点优化的大暴力。

算法 3： $40pts$

$n \le 15$

考虑DP，$f[i][s]$表示当前考虑了前i短的边，联通状态为s的方案数，s的表示方法可以使用最小表示法之类的，状态转移只需要考虑下一条边连接了哪些点即可。

算法 4： $60pts$

$n \le 30$

如果正解常数不够优秀，超时后的保底分数。事实上也可能被非常优秀的大暴力们水过去。

算法 5： $20pts$

$a_i \le n$

有用的边只剩下前n条，假设x这条边不存在最小生成树里，那么必然是使得前$x-1$条边形成的生成树上出现了环，枚举环长后组合数学随便算算即可。

算法6： $100pts$

原题数据范围

考虑优化算法3，并不需要表示出具体的连通关系，只需要知道连通块的大小即可。于是方案数变成了40的整数分拆数，只有$P(40)=37338$，枚举时也只需要枚举连通块大小而不需要枚举具体是哪个连通块，显然不同大小的连通块只会有$\sqrt{n}$个，枚举两个连通块复杂度是$O(n)$的。而需要进行枚举的转移也只有$n-1$次（只有$n-1$条有效边），其余根本不需要转移。所以最终复杂度是$O(n^2P(n))$的。