

T1

当所有数和大于 0 两人一定会优先取大的，当和小于 0 时两人都会优先取小的，排序后模拟一遍。

T2

50pts

首先对于所有音符排序，对于每个音符，暴力预处理出以该音符为一首乐曲中最低的音符能够加多少音符，然后枚举两个音符分别作为两个乐曲的最低音符，时间复杂度 $O(n^2)$ 。

100pts

用单调队列滑动窗口可以在 $O(n)$ 的时间复杂度中处理出排序后每个音符能加的音符数量。接下来只需要从后往前记录截止到每个音符位置前面所有的音符能加的音符数量的最大值，然后从前往后枚举即可。

T3

20pts

枚举所有灯点亮的方式即可。

80pts

考虑转化为最小生成树去做，若两点之间的边权高于 m 则改为 m ，之后枚举起始点。

正解

造一个虚点连接所有点，边权为 m ，从虚点连接图中的点的操作视为直接消耗 m 点星光能量点亮一盏灯，在新图上跑最小生成树即可。

T4翘课计划

- 考点：动态规划，背包问题。
- 难度：CSP-J T4

测试点 1~3

给暴力搜索留的分。

测试点4~7

留给暴力讨论的部分分。

因为 $m = 3$ ，情况很少，对于某一天你可能有这几种情况：

- 对于 001，111 等情况，小C 可以翘 1 节课，少在教学楼呆 1 小时。
- 而对于 101 的情况，小C 第一次可以选择翘 1 节课，少在教学楼呆 2 小时。

贪心的先选择少呆 2 小时的情况，剩下的只能选少呆 1 节课的情况。

测试点14~16

其实就是对上一个做法的扩展。

对于每一天，小 C 可以选择翘 1 节课，少呆 $r - l$ 个小时。

同样贪心的选择少呆时间长的课，最后如果还能翘课，那么都只能少呆 1 个小时。

测试点8~11

进一步扩展上面的做法。

对于一天，我们可以考虑每一节课选择翘或者不翘，暴力得到 $c_{i,j}$ ，表示第 i 填翘 j 节课的情况下小 C 要在教学楼里呆最少 $c_{i,j}$ 个小时。

然后就是一个分组背包。

时间复杂度为 $O(n2^m + nmk)$ 。

测试点12~13 17~20

我们不用暴力的考虑每一节课选择翘或者不翘，而只需要考虑这一天中最早的课和最晚的课是什么就可以（中间的课翘了也不会减少呆在教学楼的时间）

所以就可以枚举最早的课 j ，最晚的课 k ，计算一个要翘多少课，此时小 C 要在教学楼呆 $k - j + 1$ 小时。

这样就可以 $O(nm^2)$ 的时间复杂度求出 $c_{i,j}$ 。