1.线段（seg）

课表已知要求 k 是一个很经典的贪心问题：设置一个变量 pos 表示当前已经选取的线段最靠右的右端点位置，初值为 0，每次在所有 $l > pos$ 的线段中选出右端点最小的线段 i，选取它并令 $pos = r_i$。

基于该贪心进行 dp：设 $f_{i,j}$ 表示当前 $pos = i$，已经选了 j 条线段的方案数。考虑枚举下一条选择的线段的右端点位置 k 转移：满足 $i < l \le r = k$ 的线段有 $k - i$ 条，其至少应该有一条线段存在，贡献为 $2^{k-i} - 1$；$i < l < r < k$ 的线段都不能存在；$i < l \le k < r$ 的线段有 $(n-k) \times (k-i)$ 条，其存在与否无所谓，贡献为 $2^{(n-k)\times(k-i)}$。综合一下，也就是：

$$f_{i,j} \times (2^{k-i} - 1) \times 2^{(n-k)\times(k-i)} \to f_{k,j+1}(k > i)$$

暴力转移即可，时间复杂度 $O(n^3)$。

2.计算（calc）

直接容斥，枚举一个子集 S，其贡献就是 $(-1)^{|S|}\frac{n}{\prod a_i(a_i \in S)}$ 是 2^k 的，难以通过。

考虑 dp。设 $f_{i,j}$ 表示 $[1,i]$ 至少被一个 a 整除的数有多少个，那么答案就是 $n - f_{n,m}$。

不难写出转移：$f_{i,j} = \lfloor \frac{n}{a_i} \rfloor + f_{i,j-1} - f_{\lfloor \frac{n}{a_i} \rfloor, j-1}$。

然而直接做的复杂度是 $O(nk)$ 的，与暴力相同。

注意到，$\lfloor \frac{n}{x} \rfloor$ 的取值只有 $O(\sqrt{n})$ 个，所以有效的状态其实也只有 $O(\sqrt{n}k)$。

然而，直接用 map 或者哈希表记忆化的话空间复杂度难以接受，考虑设置阈值 lim，当 $i \le lim$ 时正常记忆化，而当 $i > lim$ 时不记忆化，时空复杂度都变得可以接受。

如果把 a 排序一下先处理大的会变快很多，但这个优化并不是本题想要考察的内容，故用于参考用时的 std 并未添加。

3.球（ball）

几乎无思维的讨论题，以弥补本场比赛偏少的代码。

把 `\` 看作 0，把 `/` 看作 1，那么问题就等价于区间翻转，再询问该区间最长的 01 串长度。

问题看起来就非常的线段树。在每个节点存储：区间长度、左侧第一个极长颜色段及其颜色，左侧第二个极长颜色段，右侧第一个极长颜色段及其颜色，右侧第二个极长颜色段，区间内最长的 01 串，区间内最长的 10 串（供翻转用）。从左右儿子讨论合并即可，具体细节课件代码。

看起来似乎细节很多，但是写起来的体验其实很好，结合样例或者对拍可以很容易的调试。

这是笔者的实现，也许有其他更简单的维护方法。

时间复杂度 $O(n + m\log n)$。

4.数列（array）

首先，数列中的数可以分为两种：作为 k 个之一加入的数和作为和加入的数。不妨称后者为“特殊数”。把构造 s 时每次取 k 个及其和并加入的操作称为"一轮"。

> 结论：$\forall i \ge 0, [i \times (k^2+1) + 1, (i+1) \times (k^2+1)]$ 这个区间中有且仅有一个特殊数，其余的数恰好进行 k 轮。

考虑归纳证明：

当 $i=0$ 时，特殊数就是 $\frac{k(k+1)}{2}$ ，显然成立。

假设已经知道了第 i 轮的特殊数是 x，其属于 $[i\times(k^2+1)+1,(i+1)\times(k^2+1)]$，那么 $[i\times k,i\times k+k-1]$ 这些轮具体选了那些数都是已知的了，对应的特殊数也就自然可求。具体的，对于第 $i\times k+t$ 轮（$t\in[0,k-1]$），其对应的特殊数应该是：

$\sum_{j=1}^{k} i(k^2+1)+j+tk+[i(k^2+1)+j+tk\geq x]$

$=k\times[i(k^2+1)+tk]+\frac{k(k+1)}{2}+\max(0,\min(k,i(k^2+1)+k+tk-x+1))$

$=(k^2+1)(ik+t)+\frac{k(k+1)}{2}-t+\max(0,\min(k,i(k^2+1)+k+tk-x+1))$

不难发现，这个式子依然属于 $[(ik+t)(k^2+1)+1,(ik+t+1)(k^2+1)]$。所以原命题成立。

有了这个结论，问题就变得简单了。设 $bel=\lfloor\frac{n-1}{k^2+1}\rfloor$ 为 n 所在的段编号，之前的式子在已知第 i 段特殊数的时候就可以递推求出 $[ik,ik+k-1]$ 中任意段的特殊数，我们可以按照类似于把 bel k 进制分解的方式求出 bel 段的特殊数 x，接下来分情况讨论：

1. $n=x$，答案就是 $(bel+1)\times(k+1)$。
2. $n\neq x$，此时 n 前面的特殊数应该有 $bel\times k+\lfloor\frac{((n-1)\%(k^2+1)-[n>x])}{x}\rfloor$ 个，而比 n 小的特殊数应该有 $bel+[x<n]$ 个，两者相减就是 n 前面且比 n 大的特殊数数量，再加 n 就是答案。

代码实现把上面的式子抄下来即可，非常简单。

瓶颈在于求 x，时间复杂度 $O(T\log\left(\frac{n}{k^2}\right))$。