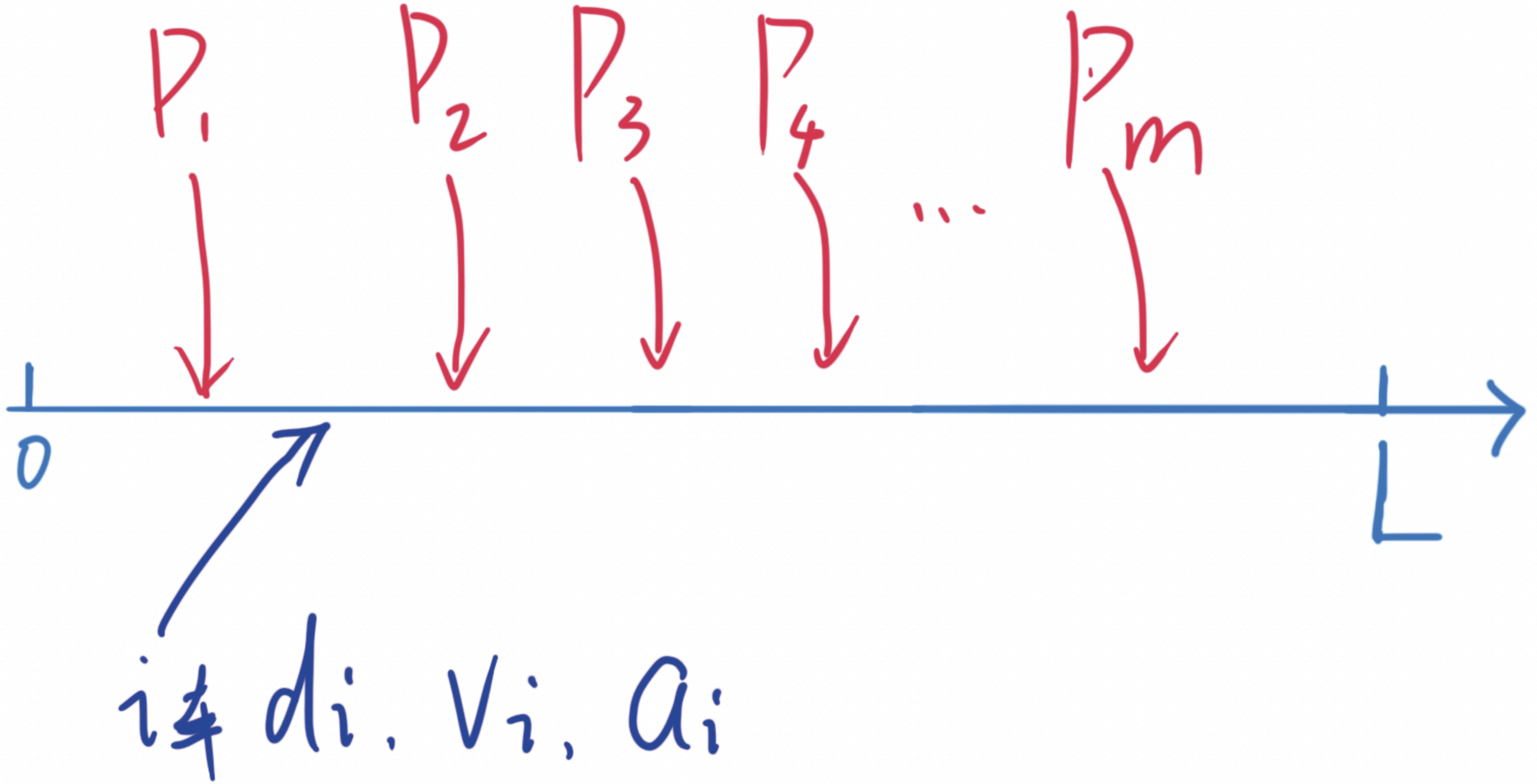
# P11232 [CSP-S 2024] 超速检测

**题意**

主干道值域 $[1, L]$，$n$ 个辆车，$m$ 个测速仪，每辆车一开始不一定在主干道上。

给定每辆车 $i$，进入主干道的起始位置 $d_i$，初始速度 $v_i$，以及加速度 $a_i$。

- Q1：有几辆车超速?
- Q2：最多能关掉多少个测速仪，需要补抓到虽有超速车辆。

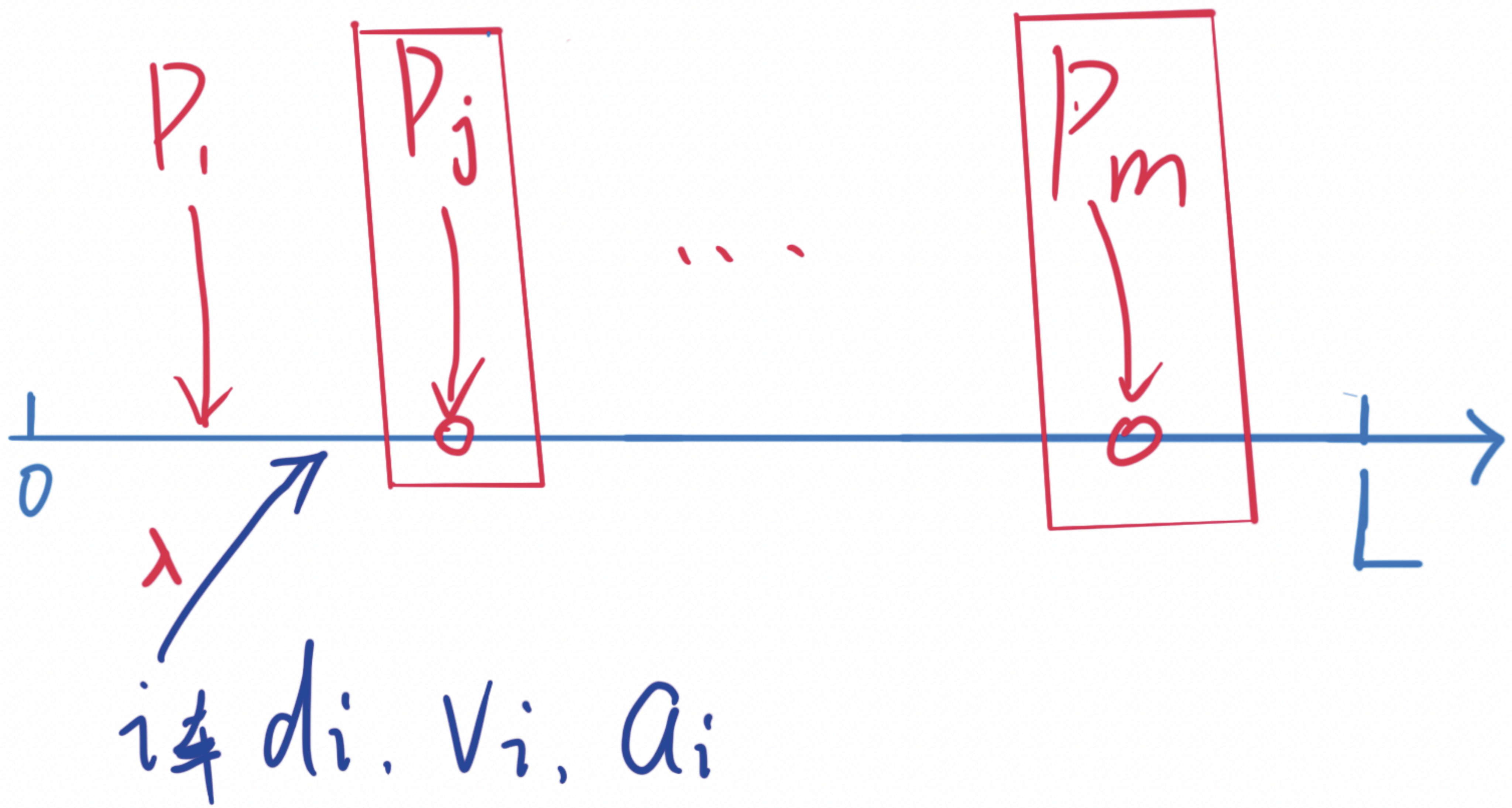

$P_1$
$P_2$
$P_3$
$P_4$
......
$P_m$
0
L
车 $d_i$, $v_i$, $a_i$

## 分析

**Subtask 1，** $1 \le n, m \le 20$；

尝试枚举 $m$ 个摄像头的状态（开/关）$2^m$，一种摄像头的开关方案。再枚举 $n$ 辆车对这 $m$ 个摄像头的逐一测试是否超速，得到一种 $2^m \cdot nm$ 的做法。

显然还过不了，这 $20$ 分需要做一些优化，从上述的枚举中每辆车都需要对 $m$ 个摄像头做一些优化。(其实这里部分分已经在引导满分做法了)。


Pi
Pj
...
Pm
0
L
x
读 di, Vi, ai

对于每辆车 $i$，有效检测的摄像头在驶入主干道的首个摄像头 $P_j$ ~ 驶出主干道的最后一个摄像头 $P_m$，有效检测：$P_j, P_j+1, ..., P_m$。

再根据题目中车在行驶过程中的速度变化的三种情况 ($a_i = 0, a_i > 0, a_i < 0$) 来看，不需要每辆车都需要 $P_j \sim P_m$ 个摄像头检测超速。

为什么？分类讨论就好了。

三种情况：

- $a = 0$，车子做匀速运动，对于摄像头 $P_j \sim P_m$ 而言，所有摄像头检测到的车速都是一样的，任意一个摄像头都可以判断，不妨设 $P_m$ 做为检测。
- $a > 0$，车子做匀加速运动，对于摄像头 $P_j \sim P_m$ 而言，$P_m$ 摄像头所检测到的车速是所有摄像头的最大值，故用 $P_m$ 检测车速是否超速即可。
- $a < 0$，车子做匀减速运动，对于摄像头 $P_j \sim P_m$ 而言，$P_j$ 摄像头所检测到的车速是所有摄像头的最大值，故用 $P_1$ 检测车速是否超速即可。

如何快速定位摄像头 $P_j$ 的位置：二分即可，例如：`Pj = lower_bound( P.begin(), P.end(), di )`。

$\exists(j \le i \le m)P_i \ge d_i \backslash \text{and} v_i \ge V$，则说明该车辆超速。

至此，你可以拿到特殊性质 $A$ 和 $C$ 的分数了，也就是 $a = 0$，以及 $a < 0$ 的情况。

如何计算车辆的瞬时速度，题目有给了 $2$ 种方式，可以到末尾查看。

**Subtask 2、3、4，** $1 \le n, m \le 10^5$；

不妨设 $f_i$：每辆车 $i$ 碰到的首个测速仪。

可以分类讨论三种情况：

- $a = 0$，车辆匀速运动，区间 $[f_i, P_m]$ 都可以检测车辆 $i$ 是否超速。
- $a > 0$，车辆匀加速运动，区间 $[f_i, P_m]$ 中可以检测到车辆 $i$ 超速，这一段肯定靠后也就是 `后缀` 。
- $a < 0$，车辆匀减速匀速，区间 $[f_i, P_m]$ 中可以检测到车辆 $i$ 超速，这一段肯定靠后也就是 `前缀` 。

可不可以，把具体检测的车速仪区间处理处理，显然是可以的。由此得到了共 $n$ 个区间。

至此，第一问答案就出来了，有几辆车超速了，等价于区间的个数。

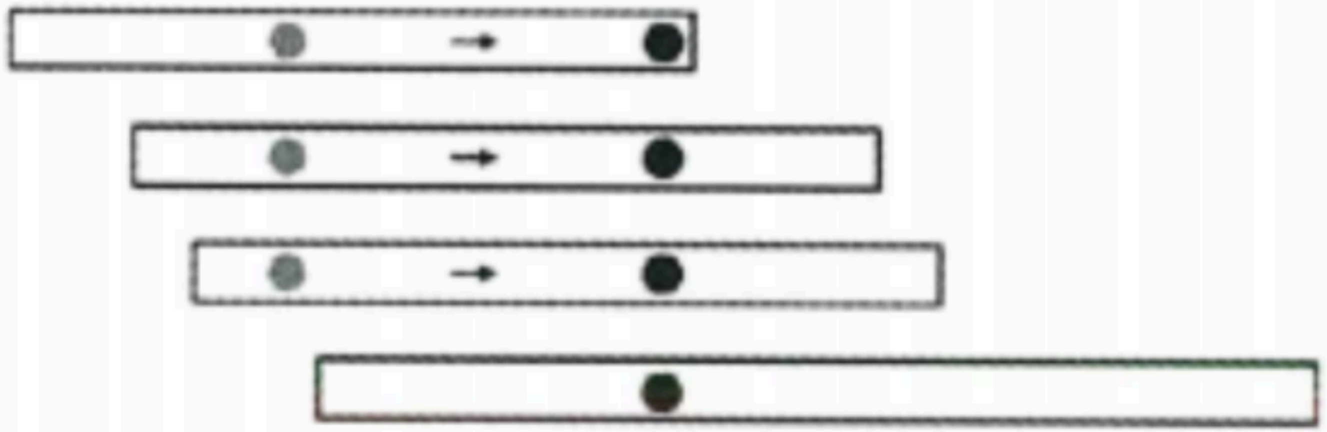
第二问：最多能关掉多少个测速仪?

贪心（`最多`），做题做多了就会有这个感觉。

问题转化：保留若干个点（测速仪）能覆盖这 $n$ 个区间，保证 $n$ 个区间至少包含一个点的情况，最少的点数。

> 入门组贪心算法的常见区间问题：区间选点问题。
>
> 给定若干个区间，每个区间内都至少一个点，不同区间可以同一个，问至少多少个点可以覆盖这些区间。
>
> 按照结束位置从小到大排序，从区间 1 到 n 进行选择：对于当前区间，若集合宏的数不能覆盖它，则将区间末尾的数加入到集合。
>
> 贪心策略：取最后一个。

如图 1—1—2 所示，如果选灰色点，则移动到黑色点会更优。



设使用 $k$ 个点即可把若干个区间就覆盖掉，**第二问的答案**就是可以关闭 $m-k$ 个摄像头。

接下来是二分的细节，check：第 $i$ 辆车走 $j$ 测速仪时是否超速，也就是 $\sqrt{V_i^2 + 2a(P_j - d_i)} > V$，是否为真。

显然根号 $\sqrt{}$ 的运算会产生精度的丢失，本题只关心两边的大小不关心具体的值，所以可以直接 `平方` 取消根号操作带来的误差。

至此完毕，更多细节请看代码注释。

## 参考代码

```
struct seg {
    int l, r;
    bool operator<(const seg &b) const { return r < b.r; }
};
vector<seg> s; /* 预处理出 n 干个检测超速区间 */
vector<int> p; /* 测速仪p */
int t, n, m, L, V;
int d[N], v[N], a[N];
/* 计算瞬时速度的平方, v0^2+2as */
int speed(int vi, int ai, int si) { return vi * vi + 2 * ai * si; }
```

```
void get(int d, int v0, int ai) {
    /* 首个测速仪下标 */
    int pj = lower_bound(p.begin(), p.end(), d) - p.begin();

    /* 不存在测速仪、匀时运动时初速度v0就已超速了  */
    if (pj >= m)
        return;
    if (ai == 0) {
        if (v0 > V)
            s.push_back({pj, m - 1});
        return;
    }

    /* 二分 */
    int l = pj - 1, r = m, pk = -1, mid;
    /* 匀减速，前缀，找最后一个满足的测速仪位置 pk，[pj, pk] */
    if (ai < 0) {
        while (l + 1 < r) {
            mid = (l + r) >> 1;
            if (speed(v0, ai, p[mid] - d) > V * V)
                l = mid, pk = l;
            else
                r = mid;
        }
        if (pk != -1)
            s.push_back({pj, pk});
    }
    /* 匀加速，后缀，找首个满足的测速仪位置 pk，[pk,pm] */
    if (ai > 0) {
        while (l + 1 < r) {
            mid = (l + r) >> 1;
            if (speed(v0, ai, p[mid] - d) > V * V)
                r = mid, pk = r;
            else
                l = mid;
        }
        if (pk != -1)
            s.push_back({pk, m - 1});
    }
}
```

```
void solve() {
    s.clear(), p.clear(); /* 多测清空 */

    cin >> n >> m >> L >> V;
    for (int i = 1; i <= n; i++)
        cin >> d[i] >> v[i] >> a[i];
    for (int i = 1, x; i <= m; i++)
        cin >> x, p.emplace_back(x);

    /* 处理若干个区间 */
    for (int i = 1; i <= n; i++)
        get(d[i], v[i], a[i]);

    /* 贪心，区间选点问题 */
    sort(s.begin(), s.end());
    int las = -1, k = 0;
    for (auto si : s)
        if (las < p[si.l])
            k++, las = p[si.r];
    cout << s.size() << " " << m - k << endl;
}
int main() {
    ios::sync_with_stdio(false), cin.tie(nullptr);
    cin >> t;
    while (t--) {
        solve();
    }
    return 0;
}
```

## 时间复杂度分析

$O(nlogn + nlogm)$

- 每辆车总处理时间：$O(logm)$
- $n$ 辆车总时间：$O(nlogm)$
- 排序：$O(nlogn)$
- 贪心选择阶段：$O(n)$
- 执行贪心选择：$O(n)$

## 主要考察考点

- 部分分引导正解。
- 耐心读题并理解含义。
- 贪心、二分。

计算车辆的瞬时速度：

- 当一辆车的初速度为 $v_0$、加速度 $a \neq 0$，做匀加速运动，则当它的位移（即行驶路程）为 $s$ 时，这辆车的瞬时速度为 $\sqrt{v_0^2 + 2 \times a \times s}$。
- 当一辆车的初速度为 $v_0$、加速度 $a \neq 0$，在它的位移（即行驶路程）为 $\frac{v_1^2 - v_0^2}{2a}$ 时，这辆车的瞬时速度为 $v_1$。