上下文管理器

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	256 megabytes

Agent 在处理长任务时，会不断向上下文窗口中加入新的消息。由于模型上下文长度有限，系统需要在 token 总数过大时压缩旧消息，或者删除最早的消息。本题中的规则是对上下文维护过程的简化抽象，并不代表真实系统的具体实现。你需要模拟这个上下文维护器。

系统维护若干条当前仍在上下文中的消息。每条消息包含三个属性：

- 编号 id；
- 当前 token 数 t；
- 重要性等级 r。

系统还有一个 token 上限 W。设当前上下文中所有消息的 token 总数为 S。每当加入一条新消息后，如果 $S > W$，系统会立刻执行维护过程。

维护过程如下：

1. 如果当前上下文中存在尚未被压缩过的消息，则选择其中一条消息进行压缩。
2. 被选择的消息需要满足：
   - 重要性等级 r 最小；
   - 若 r 相同，则编号 id 最小。
3. 若该消息当前 token 数为 x，则压缩后 token 数变为 $\lceil \frac{x}{2} \rceil$.
4. 每条消息最多只能被压缩一次。
5. 重复上述压缩过程，直到 $S \le W$，或者当前上下文中的所有消息都已经被压缩过。
6. 如果所有当前消息都已经被压缩过，但仍然有 $S > W$，则不断删除最早进入上下文的消息，直到 $S \le W$。

你需要处理 q 个操作，并在询问时输出当前系统状态，具体要求请看输出格式。

Input

输入的第一行包含两个正整数 q, W $(1 \le q \le 2 \cdot 10^5, 1 \le W \le 2 \cdot 10^{14})$ — q 表示操作个数，W 表示上下文窗口允许的最大 token 总数。

接下来 q 行，每行表示一个操作。操作共有三种。

- `ADD id t r`

  表示向上下文末尾加入一条消息。该消息的编号为 id，token 数为 t，重要性等级为 r。

  加入后，系统立刻执行维护过程。

- `ASK id`

  查询编号为 id 的消息当前是否仍在上下文中。

  若存在，输出该消息当前的 token 数；否则输出 -1。

- `STAT`

  查询当前上下文状态。

  输出两个整数 cnt, S，其中 cnt 表示当前上下文中的消息数量，S 表示当前上下文中的 token 总数。

对于所有操作，满足：$1 \le id, t, r \le 10^9$.

保证所有 `ADD` 操作中出现的 id 两两不同。

Output

对于每个 `ASK` 和 `STAT` 操作，输出一行答案。

对于 `ASK id`:

- 若编号为 id 的消息当前仍在上下文中，输出它当前的 token 数;
- 否则输出 -1。

对于 `STAT`，输出两个整数 cnt, S，其中 cnt 表示当前上下文中的消息数量，S 表示当前上下文中的 token 总数。

Example

standard input	standard output
13 10 ADD 1 6 5 ADD 2 7 1 ASK 2 STAT ADD 10 5 3 ASK 1 ADD 11 4 5 STAT ADD 12 8 5 STAT ASK 2 ADD 3 9 1 STAT	4 2 10 3 3 9 3 9 -1 2 9

Note

步骤	操作	当前上下文	说明	输出
当前上下文按消息进入顺序列出，格式为 $\{(id,t,r),\ldots\}$， 其中 id 为消息编号，t 为当前 token 数，r 为重要性等级。				
1	`ADD 1 6 5`	$\{(1,6,5)\}$	$S=6\le 10$，无需维护。	
2	`ADD 2 7 1`	$\{(1,6,5),(2,4,1)\}$	$S=13$，压缩消息 2：$7\to 4$。	
3	`ASK 2`	$\{(1,6,5),(2,4,1)\}$	消息 2 仍在上下文中。	`4`
4	`STAT`	$\{(1,6,5),(2,4,1)\}$	共有 2 条消息，$S=10$。	`2 10`
5	`ADD 10 5 3`	$\{(1,3,5),(2,4,1),(10,3,3)\}$	压缩消息 10：$5\to 3$；再压缩消息 1：$6\to 3$。	
6	`ASK 1`	$\{(1,3,5),(2,4,1),(10,3,3)\}$	消息 1 仍在上下文中。	`3`
7	`ADD 11 4 5`	$\{(2,4,1),(10,3,3),(11,2,5)\}$	压缩消息 11：$4\to 2$；随后删除消息 1。	
8	`STAT`	$\{(2,4,1),(10,3,3),(11,2,5)\}$	共有 3 条消息，$S=9$。	`3 9`
9	`ADD 12 8 5`	$\{(10,3,3),(11,2,5),(12,4,5)\}$	压缩消息 12：$8\to 4$；随后删除消息 2。	
10	`STAT`	$\{(10,3,3),(11,2,5),(12,4,5)\}$	共有 3 条消息，$S=9$。	`3 9`
11	`ASK 2`	$\{(10,3,3),(11,2,5),(12,4,5)\}$	消息 2 已被删除。	`-1`
12	`ADD 3 9 1`	$\{(12,4,5),(3,5,1)\}$	压缩消息 3：$9\to 5$；随后删除消息 10 和 11。	
13	`STAT`	$\{(12,4,5),(3,5,1)\}$	共有 2 条消息，$S=9$。	`2 9`