

Medium 1332 117 Add to List Share

Your system should accept a `timestamp` parameter (**in seconds** granularity), and you may assume that calls are being made to the system in chronological order (i.e., `timestamp` is monotonically increasing). Several hits may arrive roughly at the same time.

- `HitCounter()` Initializes the object of the hit counter system.
- `void hit(int timestamp)` Records a hit that happened at `timestamp` (**in seconds**).

Several hits may happen at the same `timestamp`.

- `int getHits(int timestamp)` Returns the number of hits in the past 5 minutes from `timestamp` (i.e., the past 300 seconds).

Input

Output

Explanation

```
HitCounter hitCounter = new HitCounter();
hitCounter.hit(1);      // hit at timestamp 1.
hitCounter.hit(2);      // hit at timestamp 2.
hitCounter.hit(3);      // hit at timestamp 3.
hitCounter.getHits(4);  // get hits at timestamp 4, return 3.
hitCounter.hit(300);    // hit at timestamp 300.
hitCounter.getHits(300); // get hits at timestamp 300, return 4.
hitCounter.getHits(301); // get hits at timestamp 301, return 3.
```

- `1 <= timestamp <= 2 * 109`
- All the calls are being made to the system in chronological order (i.e., `timestamp` is monotonically increasing).
- At most 300 calls will be made to `hit` and `getHits`.

Accepted 144,583 Submissions 216,476

Yes No

Related Topics

Similar Questions