

359. Logger Rate Limiter

Easy👍 843🔒 148❤️ Add to List🔗 Share

Design a logger system that receives a stream of messages along with their timestamps. Each **unique** message should only be printed **at most every 10 seconds** (i.e. a message printed at timestamp t will prevent other identical messages from being printed until timestamp $t + 10$).

All messages will come in chronological order. Several messages may arrive at the same timestamp.

Implement the `Logger` class:

- `Logger()` Initializes the `logger` object.
- `bool shouldPrintMessage(int timestamp, string message)` Returns `true` if the `message` should be printed in the given `timestamp`, otherwise returns `false` .

Example 1:

Input
["Logger", "shouldPrintMessage", "shouldPrintMessage", "shouldPrintMessage",
"shouldPrintMessage", "shouldPrintMessage", "shouldPrintMessage"]
[[], [1, "foo"], [2, "bar"], [3, "foo"], [8, "bar"], [10, "foo"], [11, "foo"]]

Output
[null, true, true, false, false, false, true]

Explanation
`Logger logger = new Logger();`
`logger.shouldPrintMessage(1, "foo");` // return true, next allowed timestamp for "foo" is $1 + 10 = 11$
`logger.shouldPrintMessage(2, "bar");` // return true, next allowed timestamp for "bar" is $2 + 10 = 12$
`logger.shouldPrintMessage(3, "foo");` // $3 < 11$, return false
`logger.shouldPrintMessage(8, "bar");` // $8 < 12$, return false
`logger.shouldPrintMessage(10, "foo");` // $10 < 11$, return false
`logger.shouldPrintMessage(11, "foo");` // $11 \geq 11$, return true, next allowed timestamp for "foo" is

// 11 + 10 = 21

Constraints:

- $0 \leq \text{timestamp} \leq 10^9$
- Every `timestamp` will be passed in non-decreasing order (chronological order).
- $1 \leq \text{message.length} \leq 30$
- At most 10^4 calls will be made to `shouldPrintMessage` .

Accepted 159,362 Submissions 217,566

Seen this question in a real interview before?

Yes

No

Companies🔒i^

0 ~ 6 months 6 months ~ 1 year 1 year ~ 2 years

Google | 27

Apple | 5

Microsoft | 3

Facebook | 2

Related Topics^

Similar Questions^

```
1 class Logger:
2
3     def
4         __init__(self):
5             """
6                 Initialize
7                 your data structure
8                 here.
9             """
10
11         def
12             shouldPrintMessage(self, timestamp: int, message: str) -> bool:
13                 """
14                     Returns true
15                     if the message should
16                     be printed in the
17                     given timestamp,
18                     otherwise returns
19                     false.
20
21                     If this method
22                     returns false, the
23                     message will not be
24                     printed.
25
26                     The timestamp
27                     is in seconds
28                     granularity.
29                 """
30
31         # Your Logger object
32         # will be instantiated
33         # and called as such:
34         # obj = Logger()
35         # param_1 =
36         # obj.shouldPrintMessage
37         # (timestamp,message)
```