



[BACK](#)

Regular Months

 158,354



[<](#) [DESCRIPTION](#) [MY SOLUTIONS](#) [LEADERBOARD](#) [COMMENTS](#) [README](#) [>](#)

[CODEWRITING](#) [SCORE: 300/300](#)

In an effort to be more innovative, your boss introduced a strange new tradition: the first day of every month you're allowed to work from home. You like this rule when the day falls on a Monday, because any excuse to skip rush hour traffic is great!

You and your colleagues have started calling these months *regular* months. Since you're a fan of working from home, you decide to find out how far away the nearest *regular* month is, given that the `currMonth` has just started.

For your convenience, here is a list of month lengths (from January to December, respectively):

- Month lengths: 31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31 .

Please, note that in **leap years** February has 29 days.

Example

For `currMonth = "02-2016"` , the output should be `regularMonths(currMonth) = "08-2016"` .

February of 2016 year is *regular*, but it doesn't count since it has started already, so the next *regular* month is August of 2016 - which is the answer.

Input/Output

- [execution time limit] 4 seconds (js)**
- [input] string currMonth**

A string representing the current month in the format `mm-yyyy` , where `mm` is the number of the month (1-based, i.e. 01 for January, 02 for February and so on) and `yyyy` is the year.

Guaranteed constraints:
 $1 \leq \text{int}(\text{mm}) \leq 12$,
 $1970 \leq \text{int}(\text{yyyy}) \leq 2100$.

- [output] string**

The earliest *regular* month after the given one in the same format as `currMonth` .

[JavaScript (ES6)] Syntax Tips

```
// Prints help message to the console
// Returns a string
function helloWorld(name) {
  console.log("This prints to the console when you Run Tests");
  return "Hello, " + name;
}
```