



# STAR

YOU ARE A STAR



## Preliminaries



**Language:** C

**Build:** `cc -o star *.c /path/to/our/my_putchar.c /path/to/our/main.c`

The totality of your source files, except all useless files (binary, temp files, obj files,...), must be included in your delivery.



- ✓ Don't push your `main` function into your delivery directory, we will be adding our own. Your files will be compiled adding our `main.c` and our `my_putchar.c` files.
- ✓ You are only allowed to use the `my_putchar` function to complete the following tasks, but **don't push it** into your delivery directory, and don't copy it in *any* of your delivered files.



The only allowed system call for this project is `write`.

Write a function that displays a star, based on its given size.  
If the size is 0, don't display anything (but it is not an error).

The function must be prototyped as follows:

```
void star(unsigned int size);
```

Here's some output with different sizes, it is up to you to deduce the rules regarding the star formation.



These examples (and maybe more) can also be found in a simple text files given with this subject.

In these example our star binary take a parameter which passed to the `star` function.

```
Terminal
~/B-CPE-100> ./star 1 | cat -e
  *$
*** **$
 *  *$
*** **$
  *$

~/B-CPE-100> /star 2 | cat -e
  *$
 * *$
***** **$
 *      *$
 *      *$
 *      *$
***** **$
 * *$
  *$

~/B-CPE-100> /star 5
      *
     * *
    *  *
   *   *
  *    *
 *     *
*****
*       *
*       *
*       *
*       *
*       *
*       *
*       *
*****
      *
     * *
    *  *
   *   *
  *    *
 *     *
      *
```

{EPITECH}

