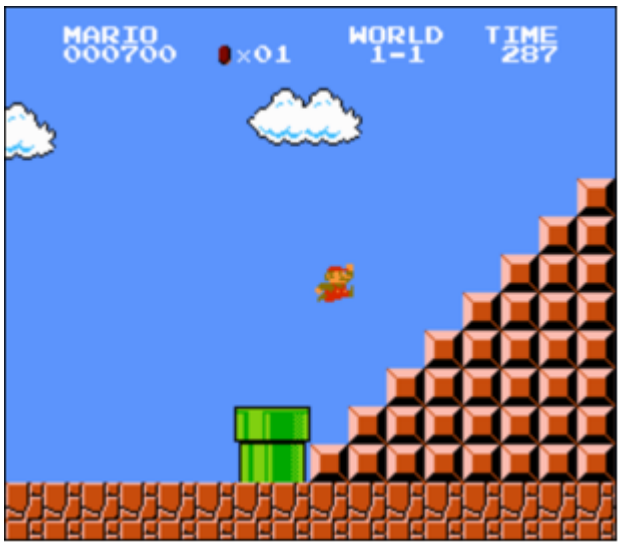
This is CS50x

OpenCourseWare

Donate (https://cs50.harvard.edu/donate)

David J. Malan (https://cs.harvard.edu/malan/)
malan@harvard.edu
(https://www.facebook.com/dmalan) (https://github.com/dmalan)
(https://www.instagram.com/davidjmalan/) (https://www.linkedin.com/in/malan/)
(https://orcid.org/0000-0001-5338-2522) (https://www.quora.com/profile/David-J-Malan)
(https://www.reddit.com/user/davidjmalan) (https://www.tiktok.com/@davidjmalan)
(https://twitter.com/davidjmalan)

Mario



Implement a program that prints out a half-pyramid of a specified height, per the below.

```
$ ./mario
Height: 4
   #
  ##
 ###
####
```

Getting Started

Log into code.cs50.io (https://code.cs50.io/), click on your terminal window, and execute `cd` by itself. You should find that your terminal window's prompt resembles the below:

```
$
```

Next execute

```
wget https://cdn.cs50.net/2021/fall/psets/6/sentimental-mario-less.zip
```

in order to download a ZIP called `sentimental-mario-less.zip` into your codespace.

Then execute

```
unzip sentimental-mario-less.zip
```

to create a folder called `sentimental-mario-less`. You no longer need the ZIP file, so you can execute

```
rm sentimental-mario-less.zip
```

and respond with "y" followed by Enter at the prompt to remove the ZIP file you downloaded.

Now type

```
cd sentimental-mario-less
```

followed by Enter to move yourself into (i.e., open) that directory. Your prompt should now resemble the below.

```
sentimental-mario-less/ $
```

Execute `ls` by itself, and you should see a `mario.py`. If you run into any trouble, follow these same steps again and see if you can determine where you went wrong!

Specification

- Write, in a file called `mario.py`, a program that recreates the half-pyramid using hashes (`#`) for blocks, exactly as you did in Problem Set 1, except that your program this time should be written in Python.
- To make things more interesting, first prompt the user with `get_int` for the half-pyramid's height, a positive integer between `1` and `8`, inclusive.
- If the user fails to provide a positive integer no greater than `8`, you should re-prompt for the same again.
- Then, generate (with the help of `print` and one or more loops) the desired half-pyramid.

- Take care to align the bottom-left corner of your half-pyramid with the left-hand edge of your terminal window.

Usage

Your program should behave per the example below.

```
$ ./mario
Height: 4
   #
  ##
 ###
####
```

Testing

While `check50` is available for this problem, you're encouraged to first test your code on your own for each of the following.

- Run your program as `python mario.py` and wait for a prompt for input. Type in `-1` and press enter. Your program should reject this input as invalid, as by re-prompting the user to type in another number.
- Run your program as `python mario.py` and wait for a prompt for input. Type in `0` and press enter. Your program should reject this input as invalid, as by re-prompting the user to type in another number.
- Run your program as `python mario.py` and wait for a prompt for input. Type in `1` and press enter. Your program should generate the below output. Be sure that the pyramid is aligned to the bottom-left corner of your terminal, and that there are no extra spaces at the end of each line.

```
#
```

- Run your program as `python mario.py` and wait for a prompt for input. Type in `2` and press enter. Your program should generate the below output. Be sure that the pyramid is aligned to the bottom-left corner of your terminal, and that there are no extra spaces at the end of each line.

```
 #
##
```

- Run your program as `python mario.py` and wait for a prompt for input. Type in `8` and press enter. Your program should generate the below output. Be sure that the pyramid is aligned to the bottom-left corner of your terminal, and that there are no extra spaces at the end of each line.

```
       #
      ##
     ###
```

```
   ###
    ####
   #####
  ######
 #######
########
```

- Run your program as `python mario.py` and wait for a prompt for input. Type in `9` and press enter. Your program should reject this input as invalid, as by re-prompting the user to type in another number. Then, type in `2` and press enter. Your program should generate the below output. Be sure that the pyramid is aligned to the bottom-left corner of your terminal, and that there are no extra spaces at the end of each line.

```
 #
##
```

- Run your program as `python mario.py` and wait for a prompt for input. Type in `foo` and press enter. Your program should reject this input as invalid, as by re-prompting the user to type in another number.

- Run your program as `python mario.py` and wait for a prompt for input. Do not type anything, and press enter. Your program should reject this input as invalid, as by re-prompting the user to type in another number.

Execute the below to evaluate the correctness of your code using `check50`. But be sure to compile and test it yourself as well!

```
check50 cs50/problems/2022/x/sentimental/mario/less
```

Execute the below to evaluate the style of your code using `style50`.

```
style50 mario.py
```

How to Submit

In your terminal, execute the below to submit your work.

```
submit50 cs50/problems/2022/x/sentimental/mario/less
```