This was CS50

Harvard Extension School (https://www.extension.harvard.edu/)
Fall 2020

Mario

World 1-1

Toward the beginning of World 1-1 in Nintendo's Super Mario Brothers, Mario must hop over adjacent pyramids of blocks, per the below.

Let's recreate those pyramids in C, albeit in text, using hashes (`#`) for bricks, a la the below. Each hash is a bit taller than it is wide, so the pyramids themselves are also be taller than they are wide.

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

The program we'll write will be called `mario`. And let's allow the user to decide just how tall the pyramids should be by first prompting them for a positive integer between, say, 1 and 8, inclusive.

Here's how the program might work if the user inputs `8` when prompted:

```
$ ./mario
Height: 8
       #  #
      ##  ##
     ###  ###
    ####  ####
   #####  #####
  ######  ######
 #######  #######
########  ########
```

Here's how the program might work if the user inputs `4` when prompted:

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

Here's how the program might work if the user inputs `2` when prompted:

```
$ ./mario
Height: 2
 #  #
##  ##
```

And here's how the program might work if the user inputs `1` when prompted:

```
$ ./mario
Height: 1
#  #
```

If the user doesn't, in fact, input a positive integer between 1 and 8, inclusive, when prompted, the program should re-prompt the user until they cooperate:

```
$ ./mario
Height: -1
Height: 0
Height: 42
Height: 50
Height: 4
   #  #
  ##  ##
 ###  ###
####  ####
```

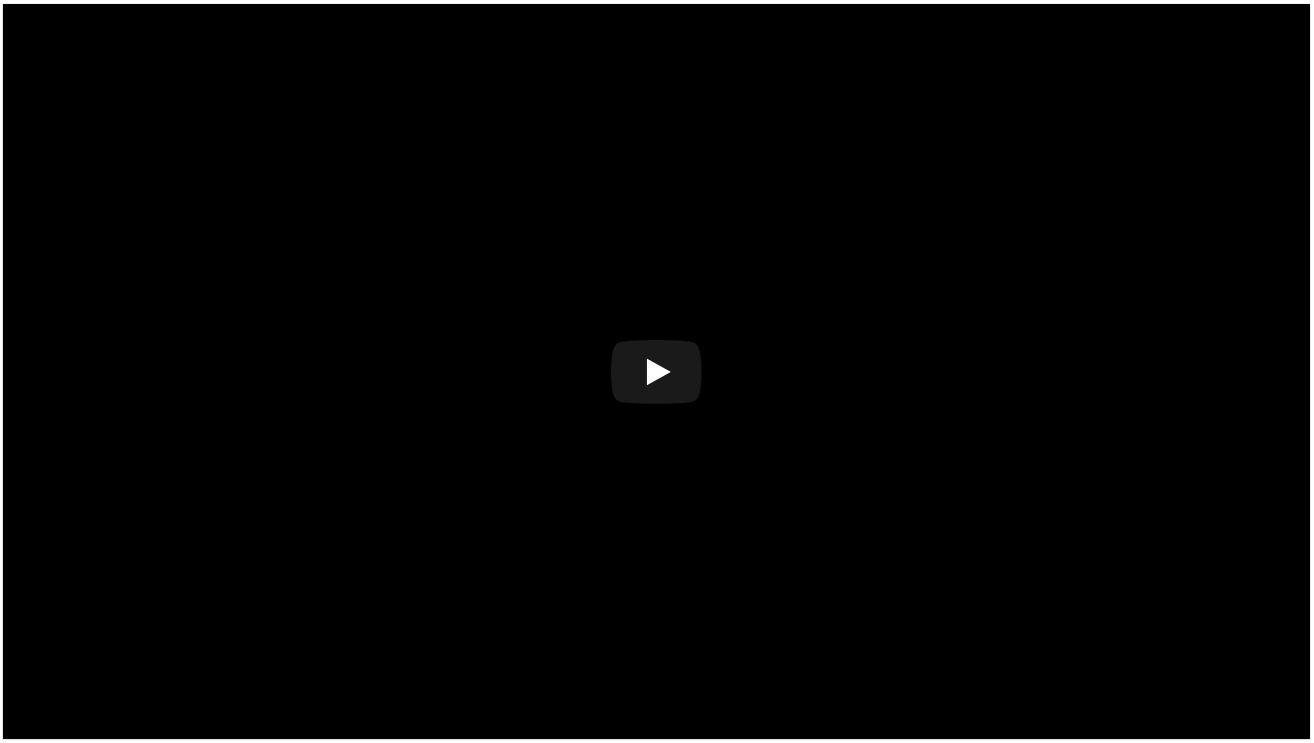
Notice that width of the "gap" between adjacent pyramids is equal to the width of two hashes, irrespective of the pyramids' heights.

Create a new directory called `mario` inside of your `pset1` directory by executing

```
~/ $ mkdir ~/pset1/mario
```

Create a new file called `mario.c` inside your `mario` directory. Modify `mario.c` in such a way that it implements this program as described!

Walkthrough



How to Test Your Code

Does your code work as prescribed when you input

- `-1` (or other negative numbers)?
- `0`?
- `1` through `8`?
- `9` or other positive numbers?

- letters or words?
- no input at all, when you only hit Enter?

You can also 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/2020/x/mario/more
```

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

```
style50 mario.c
```

How to Submit

1. Download your `mario.c` file by control-clicking or right-clicking on the file in CS50 IDE's file browser and choosing **Download**.
2. Go to CS50's Gradescope page (https://www.gradescope.com/courses/157004).
3. Click "Problem Set 1: Mario (More)".
4. Drag and drop your `mario.c` file to the area that says "Drag & Drop". Be sure it has the correct filename!
5. Click "Upload".

You should see a message that says "Problem Set 1: Mario (More) submitted successfully!" You won't see a score just yet, but if you see the message then we've received your submission!