

Table of Contents

[tl;dr](#)[Background](#)[Specification](#)[Walkthrough](#)[Usage](#)[Testing](#)[Correctness](#)[Style](#)[Staff Solution](#)[Hints](#)

Mario

tl;dr

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

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

Background

Toward the beginning of World 1-1 in Nintendo's Super Mario Brothers, Mario must hop over two "half-pyramids" of blocks as he heads toward a flag pole. Below is a screenshot.



Specification

- Write, in a file called `mario.c` in your `~/workspace/pset1/mario/more/` directory, a program that recreates these half-pyramids using hashes (`#`) for blocks.
- To make things more interesting, first prompt the user for the half-pyramids' heights, a non-negative integer no greater than `23`. (The height of the half-pyramids pictured above happens to be `4`, the width of each half-pyramid `4`, with an a gap of size `2` separating them.)
- If the user fails to provide a non-negative integer no greater than `23`, you should re-prompt for the same again.
- Then, generate (with the help of `printf` and one or more loops) the desired half-pyramids.
- Take care to left-align the bottom-left corner of the left-hand half-pyramid with the left-hand edge of your terminal window.

Walkthrough

Usage

Your program should behave per the example below. Assumed that the underlined text is what some user has typed.

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

```
$ ./mario
Height: 0
```

```
$ ./mario
Height: -5
Height: 4
  #  #
 ## ##
### ###
#### ####
```

```
$ ./mario
Height: -5
Height: five
Height: 40
Height: 24
Height: 4
  #  #
 ## ##
### ###
#### ####
```

Testing

Correctness

```
check50 cs50/2018/x/mario/more
```

Style

```
style50 mario.c
```

Staff Solution

```
~cs50/hacker1/mario
```

Hints

Try to establish a relationship between (a) the height the user would like the pyramid to be, (b) what row is currently being printed, and (c) how many spaces and how many hashes are in that row. Once you establish the formula, you can translate that to C!