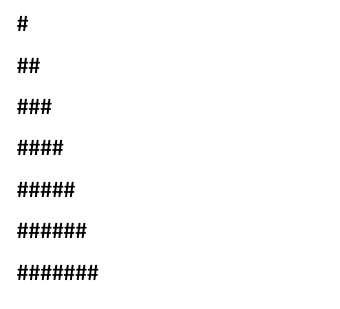
Full Stack Development Flex Week 1

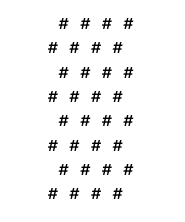
# Bonus Exercises

These exercises will provide additional practice with the concepts you have learned in the course so far this week. You will only need variables of varying data types, operators, conditionals, and loops to solve these problems. We recommend you create a new project in Visual Studio Code for this worksheet in order to continue getting practice with setting everything up!

1. Write a program that will print the following to the console using the print() function



1. Write a program that will print the following to the console using the print() function. Pay close attention to the spacing of the characters and try to match it exactly!



1. Write a program that takes a number as input from the user, counts all of the digits in that number, and prints the digit count to the console.
   1. Example: Typing 53895 will print 5 to the console
2. Find the factorial of a given number. A factorial means to multiply all of the numbers of a given number counting down to 1. For example, given the number 5, multiply 5 \* 4 \* 3 \* 2 \* 1 to get the result: 120.
3. Write a program that will allow a user to play a “guess the number” game. This program should only allow the user five guesses to guess the correct number, and display messages for the three following cases:
   1. A correct guess (which should end the program)
   2. An incorrect guess, but there are still guesses left
   3. An incorrect guess, but there are no more guesses left and the program is over. This should also display the correct number to the user
   4. BONUS: Offer a “hint” after three wrong guesses (could be a range of numbers that the number falls within, for instance)
   5. BONUS: Generate the number randomly every time the program runs.
4. Write a program that will take user input for the number of hours they have worked this week and their payrate. The output should show their total pay for the week. Be sure to account for overtime (if they work more than 40 weeks, for example, then any hours over 40 should be calculated at the standard pay rate multiplied by an overtime modifier).
   1. BONUS: Continue to ask for input for each week, adding up each week’s pay for a cumulative total until the user wants to stop.