ECE 3331 Computer Problem 2 Spring 2019
(due 11:59pm 02/19)

1) Write a C program to print the given X number pattern series. See examples below Your program should ask for a single input – the number, which will be a digit from 1-9 (if the number is not within the range, the program will terminate, e.g. if the user enters 10, or -1, the program will terminate) and then output the number and in the line below, the corresponding X pattern onto a file. The output filename is ‘CP2.txt’. Please leave 2 lines in between successive outputs for clarity. Note the appearance of the X: your output must match the examples below. Also note that your code must not hard code the output corresponding to each individual number (1-9) but rather, be general (80 points)

Example output:

```
Input N: 4

 1     1
  2   2
   3 3
    4
   3 3
  2   2
 1     1



Input N: 5

1       1
 2     2
  3   3
   4 4
    5
   4 4
  3   3
 2     2
1       1
```

2) Here, the user provides an array of characters that the programmer will show in an X pattern. The array can only be 1-9 characters long but can be any characters the user chooses. Again, see example below (20 points).

Input ! @ # $

Output

```
 !       !
 @     @
   # #
    $
   # #
@     @
!       !
```

The characters will be stored in lines in an input file. Your input file is cp2.txt. You will read in the characters from the input file – the characters will be separated by white space. An example of an input file is below (the first line has 3 characters, so your output X will consist of those three characters, the 2nd line has four characters, so your output X will consist of those four characters):

```
2 e 1
! @ # $
```

Some of the more common errors when you move from a C++ system to standard C are:

(1) using C++ style comments (//this is not standard C89)

(2) placing variable declarations AFTER executable statements (works in C++ but not standard C89)

(3) using .C or .cpp as the file extension for the source file rather than .c

Remember that your grade will depend not only on getting the right answers but also on program documentation (comments). Be sure you include the specified initial comments, describe each variable, and in the body describe each logical block. Documentation will be worth 10 (part 1) +5 (part 2) = 15 points.

Your source programs must be in two files of type .c, which you must begin with your cougarnet username followed by the letters cp2 (no spaces, all lowercase). Turn it in to Engineering Blackboard e.g. my cougarnet name is brsheth and so the zip file I will turn in will contain the following two files:

brshethcp2.c (code for part 1)

brshethcp2_2.c (code for part 2)