**HW\_6 (a,b,c)** - (3 parts)

**HW\_6a** - "Do Your Homework" - (Using a count-controlled while loop)

* Write a program that asks the user how many times he or she would like to be reminded

to do homework.

* Ask the user how many times to be reminded to do your homework.
* After the user enters a number, the program displays “Do your homework!” the number

of times.

* Use a while loop to output the phrase repeatedly.
* Make sure to include a condition that will stop the loop when the desired number of

phrases have been output.

/\* **Output:**

How many times do you need to be reminded? 5 🡨 (User enters 5)

#1: Do your homework!

#2: Do your homework!

#3: Do your homework!

#4: Do your homework!

#5: Do your homework!

Press any key to continue \*/

**HW\_6b** - “Display the alphabet”

- Use a **do-while loop** and **if statements** to complete this assignment.

* Write a program that first prompts the user:

“Do you want to see the alphabet in (U)ppercase or (l)owercase?”

* If the letter input by the user is either a U or u, then the alphabet is displayed on the

screen in uppercase letters. (see output below)

* But, if the input letter is either an ‘L’ or ‘l’, then the alphabet is displayed in lowercase letters.
* Otherwise, if the letter is not one of the above choices (U, u, L, l), a message is displayed

on the screen that tells the user: “The entry is incorrect.”

* Use an **if – else if - else** control structure to determine what to output, based on the

user’s choice.

* Include a **do-while loop** within the **if block** to output the uppercase letters.
* Likewise, within the **else if block** use a **do-while loop** to output the lowercase letters.
* Here is part of the code (if user enters U): letter = ‘A’;

while (letter <= 'Z')

* Remember that characters have numerical equivalents, so the variable, l***etter***, can

be incremented each time in the loop.

* + For example, if the variable, letter, holds an ‘A’, then letter++; increments to a ‘B’.
* Use the tab character to space the output: ‘\t’

/\* **Output**

Do you wish to see (U)ppercase or (L)owercase letters? u

A B C D E F G H I J K L M N

O P Q R S T U V W X Y Z

Press any key to continue \*/

**HW\_6c** - Minimum and maximum

Write a program that produces the output shown, based on the following information:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**/\*** **OUTPUT:**



**Write a program**:

* In *main()*, declare 2 variables: int minNumber, maxNumber;

Ask the user to enter the minimum number…………starting number

Ask the user to enter the maximum number………..ending number

Use a for loop to display every number from the starting number to the maximum number

including the minimum and maximum numbers. Also, display the square of each number.

See the above output.

**HW\_6d** - do-while loop Average 3 test score numbers

In *main()*, declare 5 variable: int score1, score2, score3;

Double average;

Char again;

In a do-while loop:

Ask the user to enter 3 test scores. Read the 3 test scores.

Display the average of the test scores.

Then the program will ask:

Does the user want to average another set? Y or N?

If the answer is yes, the program will again average a new set of numbers, and so on.

Here is the output.

