**CS10A - Sample Programming challenges introducing “loops” to begin generating various shapes…… WK10**

/\*Program generates lines of x using two types of repetition control structures

a count-controlled (a) while statement and a (b)for statement \*/

#include <iostream>

using namespace std;

int main()

{

int wcount=1, width=5;

while (wcount<=width)

{

cout<<'x';

wcount++;

}

cout<<'\n';

int line=5;

for (int colcount=0;colcount<line; colcount++)

cout<<'x';

cout<<'\n';

return 0;

}

**Practice: count-controlled loops** - Write an interactive program where two distinct “rectangle” shapes using the character 'X' maybe displayed on the screen given user input of a single positive integer no greater than 10. The user should have an opportunity to choose between the two shapes to be displayed and when to “quit” the program.

A single value (no greater than 10) should be used to represent no# of columns (or width) and no# of rows (or height) to generate the respective shapes on the screen. Using appropriate control structures (HINT! nested while loop or for statements) write in the necessary program statements that will display the desired shapes.

For example, if the user enters a positive integer 5 and chooses 1: RECTANGLE - FILLED-IN FORMAT  
the program should display the following:

XXXXX  
XXXXX   
XXXXX  
XXXXX  
XXXXX

On the other hand if perhaps the user chooses 2: RECTANGLE - OUTLINE FORMAT  
the program should display the following:

XXXXX  
X X  
X X  
X X  
XXXXX

Use meaningful variable names, nested loop or for statements with proper indentation, appropriate comments, and good prompting messages.

Write another program that asks the user for two positive integers between 2 and 10 to use for the length and width of a rectangle. If the numbers are different, the larger of the two numbers should be used for the length and smaller for the width. The program should then display a rectangle of this size on the screen using the character ’X’. For example, if the user enters either 3 8 or 8 3, the program should display the following:

XXXXXXXX  
 XXXXXXXX  
 XXXXXXXX

**Rectangle Display with variations**

Write a program that asks the user for two positive integers. The program should then use a count controlled nested loop structure to display a rectangle on the screen using the character ‘X’. The numbers entered by the user represent the “width” of each row and “height” indicating the number of rows. For example, if the user enters 5 and 7 then program should display the following:

XXXXX  
XXXXX  
XXXXX  
XXXXX  
XXXXX  
XXXXX  
XXXXX

If you are able to complete the above now here is a more advanced challenge ….

XXXXX  
X X  
X X  
X X  
X X  
X X  
XXXXX  
What type of loops did you use?

---------------------------------------------------------------------------------------------------------------  
**Triangle Display with variations – filled**

Write a C++ program that gets two inputs an integer and a character. Use repetition control structures better known as "loops" to output a filled-in triangle shape composed of the character and the width specified by the integer. If the input is an even number, it should be increased to the next odd number. Use meaningful variable names, nested loop statements with proper indentation, appropriate comments, and good prompting messages.   
For example, if the integer is 11 and the character is an asterisk (\*), the triangle shape would look like this:

|  |
| --- |
| Enter a value to represent the base of a triangle shape (not to exceed 80): 11  Enter the character to be used to generate the triangle shape (for eg., #, \* $): \*  **\***  **\*\*\***  **\*\*\*\*\***  **\*\*\*\*\*\*\***  **\*\*\*\*\*\*\*\*\***  **\*\*\*\*\*\*\*\*\*\*\***  Do you want to quit the program? (type n for no or q to quit): q |

**Triangle Display with variations – hollow**

Write a C++ program that gets two inputs an integer and a character. Use repetition control structures better known as "loops" to output a hollow triangle shape composed of the character and the width specified by the integer. If the input is an even number, it should be increased to the next odd number. Use meaningful variable names, nested loop statements with proper indentation, appropriate comments, and good prompting messages.   
For example, if the integer is 11 and the character is an asterisk (\*), the triangle shape would look like this:

|  |
| --- |
| Enter a value to represent the base of a triangle shape (not to exceed 80): 11  Enter the character to be used to generate the triangle shape (for eg., #, \* $): \*  **\***  **\* \***  **\* \***  **\* \***  **\* \***  **\*\*\*\*\*\*\*\*\*\*\***  Do you want to quit the program? (type n for no or q to quit): q |