1. Suppose the user, in response to the prompt when running the LoopTest3 program (Figure 1 below), enters 5 as the input. What would be printed on the output screen? **Note:** Show the white space, if applicable, as an underscore.

```
#include <stdio.h>
void drawLine (int size) {
        for ( ; size > 0; size--) {
                printf("*");
        }
        printf("\n");
} //drawLine()

int main()
{
        int size = 0;
        printf("Enter a number: ");
        scanf("%d", &size);
        drawLine (size);
        return 0;
}
```

Figure 1. The LoopTest3 Application

Screen output:

2. Draw a flowchart to show the operations performed in the **drawLine( )** method.

3. Rewrite the **for** statement in the **drawLine( )** method as a **while** statement.

4. Rewrite the **for** statement in the **drawLine( )** method as a **do … while** statement.

5. Suppose the user, in response to the prompt when running the LoopTest3B application (Figure 2 below), enters 6 as input. What would be printed on the output screen? **Note:** Show the white space, if applicable, as an underscore.

```
#include <stdio.h>
void drawLine (int size) {
        for ( ; size > 0; size--) {
                printf("%d", size);
        }
        printf("\n");
} //drawLine()

int main()
{
        int size = 0;
        printf("Enter a number: ");
        scanf("%d", &size);
        drawLine (size);
        return 0;
}
```

Figure 2. The LoopTest3B Application

Screen output:

6. Suppose the user, in response to the prompt when running the LoopTest3C application (Figure 3 below), enters 5 as the input. What would be printed on the output screen? **Note:** Show the white space, if applicable, as an underscore.

```
#include <stdio.h>
void drawShape (int size) {
        int temp = size;
        for (; size > 0; size--) {
                printf("%d", size);
        }
        printf("\n");
        int count = 1;
        for (; count <= temp; count++) {
                printf("%d", count);
        }
        printf("\n");
} //drawLine()

int main()
{
        int size = 0;
        printf("Enter the size of the drawing board: ");
        scanf("%d", &size);
        drawShape (size);
        return 0;
}
```

Figure 3. The LoopTest3C Application

Screen output:

7. Program Design: Solve the following problem by showing the design of your solution as a flowchart.

The problem – The program will take two integers from the user, say n1 and n2. It will then print the sum of all numbers between n1 and n2, inclusive. For example, if the user enters 5 and 9, the program will print the value of 5+6+7+8+9. On the other hand, if the user enters 5 and -2, the program will print the value of 5+4+3+2+1+0+(-1)+(-2). **Note:** Do not assume the user will always enter the smaller number first.

The flowchart –

8. **(Continued from above)** Show your design to the same problem as pseudocodes.

   The pseudocodes -

9. **(Continued from above)** Write a C program to implement your design.

10. When running the application LoopTest2, as shown in Figure 5, the user enters 4 and 5 in response to the prompt. Show the screen output. **Note:** Show the white space, if applicable, as an underscore.

```
#include <stdio.h>
void drawShape (int row, int column) {
        for (; row > 0; row--) {
                for (int temp = column; temp > 0; temp--)
                        printf("*");
                printf("\n");
        }
        printf("\n");
} //drawLine()

int main()
{
        int row = 0;
        printf("Enter the number of rows: ");
        scanf("%d", &row);
        int column = 0;
        printf("Enter the number of columns: ");
        scanf("%d", &column);
        drawShape (row, column);
        return 0;
}
```

Figure 5. The LoopTest2 Application

Screen output:

11. Suppose the programmer's original intention was to print something as shown in Figure 6 below. Revise the drawShape( ) function in the LoopTest2 program as shown in Figure 5, so it will display the correct output as shown in Figure 6. **Hint**: Display the appropriate number of white spaces for each row first, before printing the star.

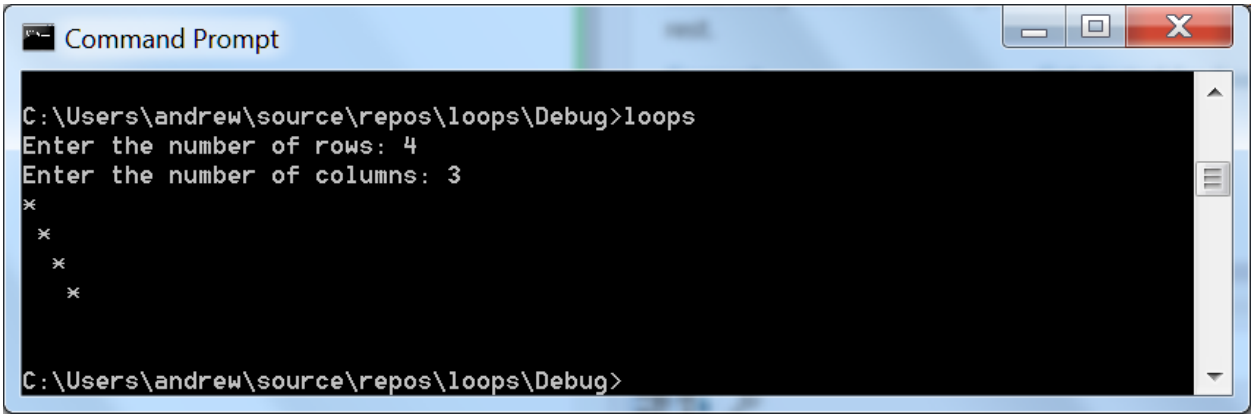

Figure 6. Screen output