### Simple Loop

1. Write a program to print the numbers 2 to 10 in increments of two. The output of your program should be 2 4 6 8 10
2. Write a program to produce a table that starts at a Celsius value of -10 and ends with a Celsius value of 60, in increments of ten degrees.
3. Write a C# program that converts gallons to liters. The program should display gallons from 10 to 20 in one-gallon increments and the corresponding liter’ equivalents. Use the relationship that 1 gallon contains 3.785 liters.
4. Write a C# program that converts feet to meters. The program should display feet from 3 to 30 in three-foot increments and the corresponding meter equivalents. Use the relationship that 1 meter is equivalent to 3.28 feet.
5. Write a C# program to convert Celsius degrees to Fahrenheit. The program should request the starting Celsius value, the number of conversion to be made, and the increments between Celsius values. The display should have appropriate headings and list the Celsius value and the corresponding Fahrenheit value. Use the relationship: Fahrenheit = (9.1) / 5.0) \* Celsius + 32.0.
6. Modify the program written in Exercise 5 to request the starting Celsius value, the ending Celsius value, and the increment Thus, instead of the condition checking for a fixed count, the condition checks for the ending Celsius value
7. Write a Program to compute the total of eight numbers
8. Write a Program to compute the average of ten numbers which will be entered by the user.
   1. For the following program determine the total number of items displayed. Also determine the first and last numbers printed.

public static void Main()   
{  
 int num = 0;  
 while (num <= 20)   
 {  
 num++;  
 Console.Write(num);   
 }  
}

* 1. Enter and run the program from Exercise 3a on a computer to verify your answers to the exercise. (a)
  2. How would the output be affected if the two statements within the compound statement were reversed (that is, if the Console.Write statement call were made before num++ statement?

1. Write a program to produce a table of the numbers 0 through 20 in increments of 2, with their squares and cubes.
2. Write a program to produce a table of numbers from 10 to 1, with their squares and cubes.
3. Write a C program that continuously requests a grade to be entered. If the grade is less than   
   0 or greater than 100, your program should print an appropriate message informing the user that an invalid grade has been entered, else the grade should be added to a total. When a grade of 999 is entered the program should exit the repetition loop and compute and display the average of the valid grades entered.
4. Write and run a C# program that calculates and displays the amount of money available in a bank account that initially has $1,000 deposited in it and that earns 8 percent interest a year. Your program should display the amount available at the end of each year for a period of ten years. Use the relationship that the money available at the end of each year equals the amount of money in the account at the start of the year plus .08 times the amount available at the start of the year.
   1. Modify the program written for Exercise 13 to initially prompt the user for the amount of money initially deposited in the account.
   2. Modify the program written for Exercise 10 to initially prompt the user for both the amount of money initially deposited and the number of years that should be displayed.
   3. Modify the program written for Exercise 10 to initially prompt for the amount of money initially deposited, the interest rate to be used, and the number of years to be displayed.
5. A machine purchased for $28,000 is depreciated at a rate of $4,000 a year for seven years. Write and run a C program that computes and displays a depreciation table for seven years, The table should have the form:

END-OF-YEAR ACCUMULATED  
YEAR DEPRECIATION VALUE DEPRECIATION  
---- ------------ ------------ ------------  
 1 4000 24000 4000  
 2 4000 20000 8000  
 3 4000 16000 12000  
 4 4000 12000 16000  
 5 4000 8000 20000  
 6 4000 4000 24000  
 7 4000 0 28000

### Nested Loops

1. Write a program to create the following figures:

-----\*

----\*\*\*

---\*\*\*\*\*

--\*\*\*\*\*\*\*

-\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*

\*

\*

\*

\*

\*

\*

\*

1. Write a program to create the following figures:

\*

\*

\*

\*

\*

\*

\*

\*\*

\*\*\*

\*\*\*\*

\*\*\*\*\*

\*\*\*\*\*\*

1. Write a program to create the following figures:

\*\*\*\*\*\*

\*\*\*\*\*\*

\*\*\*\*\*\*

\*\*\*\*\*\*

\*\*\*\*\*\*

\*\*\*\*\*\*

\*\*\*\*\*\*

\*\*\*\*\*\*

\*\*\*\*\*\*

\*\*\*\*\*\*

\*\*\*\*\*\*

\*\*\*\*\*\*

1. Write a program to create the following figures:

\*

\*\*\*

\*\*\*\*\*

\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*

\*\*\*\*\*

\*\*\*

\*

\*

\* \*

\* \*

\* \*

\* \*

\*\*\*\*\*\*\*\*\*\*\*

1. Print the decimal, octal, and hexadecimal values of all characters between the start and stop characters entered by a user. For example, if the user enters an a and z, the program should print all the characters between a and z and their respective numerical values. Make sure that the second character entered by the user occurs later in the alphabet than the first character. If it does not, write a loop that repeatedly asks the user for a valid second character until one is entered.
   1. A bowling team consists of five players. Each player bowls three games. Write a C program that uses a nested loop to enter each player’s individual scores and then computes and displays the average score for each bowler. Assume that each bowler has the following scores:

1st bowler: 286 252 265   
2nd bowler: 212 186 215   
3rd bowler: 252 232 216   
4th bowler: 192 201 235   
5th bowler: 186 236 272

* 1. Modify the program written for Exercise 6a to calculate and display the average team score. (Hint: Use a second variable to store the total of all the players’ scores.)

1. Rewrite the program written for Exercise 6a to eliminate the inner loop To do this, you have to input three scores for each bowler rather than one at a time. Each score must be stored in its own variable name before the average is calculated.
   1. Write a program to reverse the digits of a positive integer number. For example, if the number 8735 is entered, the number displayed should be 5378. (Hint: Use a do statement and continuously strip off and display the units digit of the number. If the variable num initially contains the number entered, the units digit is obtained as (num % 10). After a units digit is displayed, dividing the number by 10 sets up the number for the next iteration. Thus, (8735 % 10) is5and (8735 / 10) is 873. The do statement should continue as long as the remaining number is not zero).
   2. Run the program written in Exercise 3a on a computer, and verify the program using appropriate text data.

### Exercises 53

1. Determine the output of the following program.

int main()   
{   
 int i;   
 for (i = 20;i<= 0; i -= 4)  
 Console.WriteLine(i);  
 return 0;  
}

1. Four experiments are performed, each consisting of six test results. The results for each experiment are given below. Write a program using a nested loop to compute and display the average of the test results for each experiment.

1st experiment results: 23.2 31.5 16.9 27.5 25.4 28.6   
2nd experiment results: 34.8 45.2 27.9 36.8 33.4 39.4   
3rd experiment results: 19.4 16.8 10.2 20.8 18.9 13.4   
4th experiment results: 36.9 39.5 49.2 45.1 42.7 50.6

1. Modify the program written for Exercise 14 so that the number of test results for each experiment is entered by the user. Write your program so that a different number of test results can be entered for each experiment.
2. Write a C# program that calculates and displays the yearly amount available if $1,000 is invested in a bank account for 10 years. Your program should display the amounts available for interest rates from 6 percent to 12 percent inclusively, at 1 percent increments. Use a nested loop, with the outer loop having a fixed count of 7 and the inner loop a fixed count of 10. The first iteration of the outer loop should use an interest rate of 6 percent and display the amount of money available at the end of the first 10 years. In each subsequent pass through the outer loop, the interest rate should be increased by 1 percent. Use the relationship that the money available at the end of each year equals the amount of money in the account at the start of the year plus the interest rate times the amount available at the start of the year.