

Assignment 4 – Due in Week 10

Solve using a **for** and a **while** loop

Once you are done with your programs, upload them to eCentennial under Assessments / Assignment / Assignment 4

Assignment: (Solve using a for and a while loop). Demonstration is due at the beginning of your next-next class. You may code both loops in the same project.

1. Write a program to produce a table of numbers from 10 to 1, with their squares and cubes. Again, your table must look professional!
2. Write a program that displays a conversion table from Fahrenheit to Celsius. The program must request the starting Fahrenheit value, the ending Fahrenheit value, and the increment. Thus, instead of the condition checking for a fixed count, the condition checks for the ending Fahrenheit value. Your table must look professional!
3. 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.

Your output should be formatted as shown below

Year	Balance
1	\$1,080.00
2	\$1,166.40
3	\$1,259.71
4	\$1,360.49
5	\$1,469.33
6	\$1,586.87
7	\$1,713.82
8	\$1,850.93
9	\$1,999.00
10	\$2,158.92

4. Write a C program that continuously requests a grade. 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.

Try to use the **break** and **continue** keywords.

If the grade is 999, the program should exit the loop and display the sum, number of valid grades and the average of the valid grades entered. (You must NOT display an invalid message) P.S. The value 999 is use to terminate the cycle and should not be used in any of the calculation. You must not display an "Invalid grade"

5. 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 a 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.

Your output should be formatted as shown below

Letter	Decimal	Octal	Hex
a	97	141	61
b	98	142	62
c	99	143	63
d	100	144	64
e	101	145	65

You need separate loops to read the start and stop character and to display the table.

to obtain an int from a char use an explicit cast

to obtain an octal from an int use **Convert.ToString(«number_value», 8)**

to obtain a hexadecimal from an int use **Convert.ToString(«number_value», 16)**

6. Write a program to compute the (x, y) pairs for the equation $y = 2x^2 - x - 6$ for x in the range 1 to 5 in 0.5 increments.

x	$2x^2$	-x	-6	y
1.0	2.0	-1.0	-6	-5.0
1.5	4.5	-1.5	-6	-3.0
2.0	8.0	-2.0	-6	0.0
2.5	12.5	-2.5	-6	4.0

7. 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) is 5 and (8735 / 10) is 873. The do statement should continue as long as the remaining number is not zero).