# WORKSHEET #7

## Assumes: Ch1, Ch2, Ch3, Ch5, Ch6

1. Write a program that gets an integer value given by the user and displays, using asterisks, a filled diamond of the given side length. For example, if the side length is 4, the program should display:

\*

\*\*\*

\*\*\*\*\*

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

\*\*\*\*\*

\*\*\*

\*

1. Let n be a positive integer consisting of up to 10 digits. Write a program to list in one column each of the digits in the number n. The right-most digit should be listed at the top of the column. For example if the input number n was 170498 the program should output:

8

9

4

0

7

1

The given number 170498 has 6 digits.

HINT: if n is 3704, what is the value of the digit when computed using: digit = n%10;

1. Radioactive decay of radioactive materials can be modeled by the equation , where *A* is the amount of the material at time *t*; *AƟ* is the amount at time *Ɵ*; and *h* is the half-life. Technetium-99 is a radioisotope that is used in imaging of the brain. It has a half-life of 6 hours. Your program should display the relative amount *A/AƟ* in a patient body every hour for 24 hours after receiving a dose.
2. *The Monty Hall Paradox*. Marilyn vos Savant described the following problem in a popular magazine: *“Suppose you are on a game show and you are given the choice of three doors: Behind one door is a car; behind the others, goats. You pick a door, say No. 1, and the host, who knows what is behind the doors, opens another door, say No.3, which has a goat. He then says to you, “Do you want to pick door No.2?” Is it to your advantage to switch your original choice?”* Ms vos Savant proved that it is to your advantage, but many of her readers, including some mathematics professors, disagreed, arguing that the probability would not change because another door was opened. Your task is to simulate this game show. In each iteration, randomly pick a door number between 1 and 3 for placing the car. Randomly have the player pick a door. Randomly have the game show host pick a door having a goat (but not the door that the player picked). Increment a counter for strategy 1 if the player wins by switching to the host’s choice, and increment a counter for strategy 2 if the player wins by sticking with the original choice. Run 1000 iterations and print both counters)
3. *Mean and standard deviation*. Write a program that reads a set of floating-point data from a file (assume one data per line). When all the values have been read, print out the count of the values, the average, and the standard deviation: s = ; you can compute this quantity by keeping track of the count, the sum, and the sum of squares as you process the input values.
4. One way to scramble the letters of a message is to separate the message into two groups of characters (strings): one string composed of the even-numbered characters and the second string composed of the odd-numbered characters. Next if we create a third string that is concatenation of the two created strings, the result forms the encrypted version of the original message. This technique is called transposition cipher. Your task it to create encrypted message using the transposition cipher method.
5. *Retrieve a message from the keyboard*
6. *Using a for loop go over the input message one character at the time (see Lecture Notes Chapter 6 "Processing a String" slide)*
   1. *if the current index is even append the character to the even-numbered-characters-string*
   2. *else append the character to the odd-numbered-characters-string*
7. *Concatenate the two strings together to form the encrypted message (another string)*
8. *Display the results*

##### Sample Run#1:

Enter the message to be encrypted

  This is my top secret, I love programming!

Processing input: "This is my top secret, I love programming!"

Results:

--> String with all letters at even index is: "Ti sm oert  oepormig"

--> String with all letters at odd index is: "hsi ytpsce,Ilv rgamn!"

--> "This is my top secret, I love programming!" encrypted is: "Ti sm o ert  oepormighsi ytpsce,Ilv rgamn!"

##### Sample Run#2:

Enter the message to be encrypted

  abcdefg

Processing input: "abcdefg"

Results:

--> String with all letters at even index is: "aceg"

--> String with all letters at odd index is: "bdf"

--> "abcdefg" encrypted is: "acegbdf"

1. Often on a web page, the user is asked to supply personal information, such as a telephone number. Your program should take an input from the keyboard representing a valid telephone number. We will consider that the input is a valid telephone number if it contains exactly 10 digits and any number of dash (-) and white space characters. Keep prompting the user for a telephone number until the user gives you a valid one. Once you have a valid telephone number, you should assume that the digits (only the digits!) in the telephone number may have been encrypted by shifting each number by a constant value. For instance, if the shift is 2, a 0 becomes a 2, a 1 becomes a 3, a2 becomes a 4, … , an 8 becomes a 0 , and a 9 becomes a 1. However, we know that the user is from New York where the decrypted area code (after the shift is applied), represented by the first three digits of the input, is 212. Your program needs to decrypt the telephone number and output the decrypted telephone number with the format 212-xxx-xxxx, as well as the shift value of the encryption. If there was an error in the input and the area code cannot be decrypted to 212, you should output that information.
2. ☺