CSCI 11032/CTEC 11052 – Structured Programming I

Assignment

Submit on or before 5th December 2022

1. Following is the specifications for simulating an automatic stamp vending machine
   (a) It should dispense 25, 15 and 10 rupees worth stamps
   (b) It should accept 50, 25, 10 and 5 rupees' coins.
   (c) It can accept not more than one coin for each transaction.
   (d) If more than one coin of the same denomination is to be returned as change after dispensing the stamp, the machine cannot do it. Instead the coin should be returned and a 'no change' signal turned on.
   (e) The machine should dispense the stamp and the right change and must indicate exceptional cases such as 'insufficient amount tendered', 'no stamp available', 'no change available', etc.

   Write a program to simulate the machine. The input to the program would be: Amount tendered and the stamp requested (only one stamp). The output of the program should be whether stamp is dispensed or not, the value of the stamp dispensed, the denomination of the coins returned (if any) and no change signal if no change is returned and no stamp if the stamp is not available. The program should execute for a set of data until the user wants to quit. After each transaction the available quantity (i.e. how many) of stamps and coins should be displayed.

   Note: You have to first store the quantity (i.e. how many) of stamps and the coins that are available in the machine. You can do this by assigning constant values to each stamp and each coin given in the question. Once a transaction is done make sure to decrease quantity of the sold stamp and to increase the quantity of the respective coin (i.e. depending on the amount tendered) tendered and to decrease the quantity of the coin if that coin is returned as change to the user. These updated quantities should be considered for the next transaction. Use a number of test data and test your program.

2. Consider the following table **A** and the list **X**. The table **A** has **m** rows and **n** columns. The list X has **n** elements.

A =

1	2	3	4	5	6	7	8
2	3	4	5	6	7	8	9
3	4	5	6	7	8	9	10
4	5	6	7	8	9	10	11
5	6	7	8	9	10	11	12
6	7	8	9	10	11	12	13

X=

1
-8
3
-6
5
-4
7
-2

It is necessary to generate a new list Y which is formed by carrying out the following operations.

Y[1] = A[1][1]*X[1] + A[1][2]*X[2] + ……+ A[1][n]*X[n]
Y[2] = A[2][1]*X[1] + A[2][2]*X[2] + ……+ A[2][n]*X[n]
……..
……..
Y[m] = A[m][1]*X[1] + A[m][2]*X[2] + ……+ A[m][n]*X[n]

Write a program to read the table A and list X and display the table A and list X, followed by the values of the elements of Y.

Instructions

Read the given questions carefully before you start working on this assignment. This is an individual assignment. You are expected to complete the assignment individually. However, you can discuss the concepts with your classmates. Late assignments will be penalized 10% per day (this includes week ends as well).

Any form of notification for inability to submit an assignment including medical certificates will not be entertained. This policy is strictly enforced. Assignments should be submitted on or before the date mentioned on the assignment. If plagiarism or cheating is identified a **zero mark** will be given for the assignment to all students involved. This policy is strictly enforced. If you do have any problem or queries regarding the assignment, you can meet the lecturer and discuss.

Your programs should be creative and interactive. The program should be intended. Marks will be allocated for this.

You should upload the assignments in two files. One file for each question. The uploaded file should be a .C file or a notepad file. You should upload your program to the created folder in Moodle before midnight 5th December 2022.