BİL 102 – Computer Programming
HW 05

Last Submission Date: March 26, 2014 – 09:00

Notes:
1. For this homework only, you will provide **a separate file for each part**.
2. Because your code may be tested automatically by using test software, **strictly** obey defined I/O format in all inputs (from file or console) and file outputs and also tag declarations. You can (and should) inform user by console output in any reasonable format.

1. **(60 Pts)** In this part you will write a complete C program to handle a customer account of an investment company. It will consider 3 investment instruments and TL as shown in Table 1.

Instrument	Instrument Code
TL	L or l
USD	D or d
Gold	G or g
Investment Fund	F or f

Table 1: Investment Instruments

Your program will take the initial amount of these instruments in the user account and make some user-demanded operations (defined in Table2) on them. All amounts of instruments will be integers, resulting amounts after each operation will be rounded to the nearest integer.

Operation	Operand 1	Operand 2	Operand 3	Operation Code	Explanation
Transaction	Instrument to Sell	Instrument to Buy	Amount of Sell	T or t	Sell an instrument at a specified amount and buy an instrument instead.
Buy	Instrument to Buy	-	Amount of Buy	B or b	Buy an instrument
Sell	Instrument to Sell	-	Amount of Sell	S or s	Sell an instrument
Input	Instrument to Increase	-	Amount of Increase in Instrument	I or i	Increase the amount of an instrument in the account
Output	Instrument to Decrease		Amount of Decrease in Instrument	O or o	Decrease the amount of an instrument in the account
Report to File	-	-	-	F or f	Print current amounts of all instruments to a predefined stream
Report to Console	-	-	-	P or p	Print current amounts of all instruments to the console

Exit	-	-	-	E or e	Terminates the program

Table 2: Operations in the user interface

User Interface

- Your code will obtain initial amounts of instruments from a text file “**Initials.txt**” having the following format:

  <SWSC><I1:><AI1><SWSC><I2:><AI2><SWSC><I3:><AI3><SWSC><I4:><AI4>
  where
  Ix stands for Instrument x, where x is the code of the instrument,
  AIx stands for amount of instrument x,
  SWSC stands for some white space characters.
  An example of this file is:

  ```
           L:1500          g:20    F:0

             D:800
  ```

- Prices of the instruments will be obtained from a test file “**Rates.txt**” having the following format:

  <SWSC><I1:><PI1><SWSC><I2:><PI2><SWSC><I3:><PI3>

  where Pix represents the price(with respect to TL) of instrument x (x is different from TL)
- Name of the report file (to be produced after “Report to File” operation) will be “**Portfolio.txt**” and its format will be as follows:
  <L:><Amount of TL in the account>
  <D:><Amount of USD in the account >
  <G:><Amount of Gold in the account >
  <F:><Amount of Investment Fund in the account >
- Your program will perform the following operations until the user chooses the exit operation:
  - Take an operation code from user
  - Take all required operands of the operation (TL is not a legal instrument to be used in transaction operations in the user interface)
  - Realize the operation

  These data will be taken step by step, informing user before each input.
- Your code should also support batch mode of operation in which:
  - Input is redirected from a data file, where each operation is defined in a separate line where operands are delaminated by one space character. For example if the name of your executable file is “exec”, and the name of the data file to be redirected is “Operations.txt”, your code should be able to be executed as
    “./exec < Operations.txt”
    without any problem. An example of an operation file is below:

```
B D 20
T d g 10
f
P
I l 3000
E
```

Implementation Details

Implement the following functions with exact given names and arguments. Ordering of the arguments is also important.

- **getPrices**: gets the prices of instruments from an input file
  priceFile :(input) FILE* showing the pricing file
  dolarP: (output) price of 1 dolar
  goldP: (output) price of 1 unit gold
  fundP: (output) price of 1 fund
- **getInitials**: gets the initial amounts of instruments from an input file
  initFile :(input) FILE* showing the initials file
  dolarA: (output) amount of 1 dolar
  goldA: (output) amount of 1 unit gold
  fundA: (output) amount of 1 fund
- **transaction**: handles transaction, buy and sell operations. If the demanded amount of operation exceeds the amount of the selling item, operation is partially performed. Updates amounts.
  amoOSI: (Input / Output) amount of selling instrument
  amoOBI :(Input / Output) amount of buying instrument
  amount: (Input / Output) amount of operation
  rate: (Input) answer of "how many buying instruments can be taken by selling 1 selling instrument"
  Return Value: amount of operation normally, and a negative error code on error.
- **saveReport**: saves a report to a text file.
  reportFile. (input) FILE* showing the report file
  amoOLira: (Input) amount of TL in the portfolio
  amoODolar: (Input) amount of dolar in the portfolio
  amoOGold: (Input) amount of gold in the portfolio
  amoOFund: (Input) amount of fund in the portfolio

2. **(60 Pts)**

- Write a function "**drawReccurOf2Triangles**" that prints a pattern like the one given below. As you see, the shape is formed by horizontal recurrences of a pattern with one upper and one lower triangles. The function will get the following parameters in the given order:
  m: the height of the pattern (3 for the example)
  n: the number of empty columns between each triangle (2 for the example)
  k: number of recurrence (2 for the example)
  c: the character (x for the example)
  e: the number of empty columns before start of the pattern (0 for the example)

This function should return the number of characters it prints normally, and a negative error code on error.

x	x	x	x	x			x			x	x	x	x	x			x		

	x	x	x			x	x	x			x	x	x			x	x	x	
		x			x	x	x	x	x			x			x	x	x	x	x

- Write a function "**drawReccurOf4Triangles**" that prints a pattern like the one given below. As you see, the shape is formed by horizontal and vertical recurrences of a pattern with 2 upper and 2 lower triangles. The function will get the following parameters in the given order:
  - **rH:** horizontal recurrence of the pattern (1 for the example)
    rV: vertical recurrence of the pattern (2 for the example)

x	x	x	x	x			x							
	x	x	x			x	x	x						
		x			x	x	x	x	x					
			x	x	x	x	x		x					
				x	x	x		x	x	x				
					x		x	x	x	x	x			
x	x	x	x	x			x							
	x	x	x			x	x	x						
		x			x	x	x	x	x					
			x	x	x	x	x		x					
				x	x	x		x	x	x				
					x		x	x	x	x	x			

This function should return the number of characters it prints normally, and a negative error code on error. You will use the drawReccurOf2Triangles function in this part.

General:

1. Obey honor code principles.
2. **Read your homework <u>carefully</u>** and follow the directives about the I/O format (data file names, file formats, etc.) and submission format **<u>strictly</u>**. Violating any of these directives will be penalized.
3. Obey coding convention.
4. Do not forget to put the required **tags** in the main function.
5. Your submission should include the following file **<u>and NOTHING MORE</u>** (no data files, object files, etc):

   HW05_<StudentName>_<StudentSirname>_<student number>_part1.c
   HW05_<StudentName>_<StudentSirname>_<student number>_part2.c

   Do **NOT** compress the files you submit.
6. Do not use non-English characters in any part of your homework (in body, **file name**, etc.).
7. Deliver the printout of your work **until the last submission date**.