

## 2 March 2012 -- Computer Architectures -- part 1/2

Name, Matricola .....

A cell phone provider is offering connections charged according to duration and time of usage. It is required to write a 8086 assembly program to compute the duration of a connection and the costs to be charged.

Input data:

RECORD DB 3 DUP (?)

RATES DW 2 DUP (?)

Output data:

DURATION\_OF\_CONNECTION DW (?)

COST\_TO\_BE\_CHARGED DW (?)

Format of input data

- RECORD stores on 5 bits the starting hour, on 6 bits the starting minutes, on 1 bit the same/next day information, on 5 bits the end hour, on 6 bits the end minutes, i.e.:

X <starting hour> <starting minute> D <ending hour> <ending minute>  
1        5                          6        1        5                          6

with X = unused, D=0 stands for same day end of connection, D=1 connection ending the next day, <starting hour>, <ending hour> = 0..23; <starting minute>, <ending minute> = 0..59

It is known in advance, that the duration of the connection is at least 1 minute and at most 23 hours and 59 minutes.

X<13><35>0<15><17> stands for a connection starting at 13.35 and ending at 15.17 of the same day

X<10><22>1<9><46> stands for a connection starting at 10.22 and ending at 9.46 of the following day

- Each element of RATES stores the window hours of applicability of a rate. All elements of RATE cover the 24 hours of a day, i.e. from 00:00 to 24:00, with the following format:

X<starting hour of window><end hour of window><cost per minute>  
1            5                                  5                                  5

with the 5 bits of the cost per minute are all fractional bits, i.e.: 0.<cost per minute>. For example:

X<18><9><0,125> corresponds to the rate of 0,125 that is valid from 18:00 to 9:00 (of the next day)

X<9><18><0,375> corresponds to the rate of 0,375 that is valid from 9:00 to 18:00 (of the same day)

Format of output data

- DURATION\_OF\_CONNECTION stores on 5 + 6 bits the total duration in hours and minutes of the connection:

XXXXX <number of hours of connection> <number of minutes of connection>  
1 1 1 1 1                                  5                                  6

XXXXX<17><29> stands for a connection that lasted 17 hours and 29 minutes.

- COST\_TO\_BE\_CHARGED stores on 11 integer and 5 fractional bits the cost of the connection according to duration, windows and corresponding rates.

<68><0,475> stands for a connection that is charged 68,475 money units.

## 2 March 2012 -- Computer Architectures -- part 1/2

Name, Matricola .....

Items with corresponding points for each completed item (as incompleting items could be not evaluated)

- Item 1. = computation of the duration of the connection, and the total cost assuming that there exists just one flat rate for the 24 hours: up to 22 points;
  - e.g.: connection from 9:03 to 11:00, flat rate 0.125 per minute
- Item 2. = computation of the duration of the connection, and the total cost assuming that there exist two rate windows not crossing two days (i.e. within the same day) and that the connection as well does not cross two days (i.e. within the same day): up to 25 points;
  - e.g.: connection from 8:55 to 11:20, two rates: from 00:00 to 9:00 0.125 per minute, from 9:00 to 24:00 0.250 per minute
- Item 3. = computation of the duration of the connection, and the total cost assuming that there exist two rate windows not crossing two days (i.e. within the same day) and that the connection can cross two days: up to 28 points;
  - e.g.: connection from 8:55 to 11:20, two rates: from 00:00 to 9:00 0.125 per minute, from 9:00 to 24:00 0.250 per minute
  - e.g.: connection from 22:59 to 1:12, same two rates as previous line
- Item 4. = computation of the duration of the connection, and the total cost assuming that there exist two rate windows (with both rates and windows that can cross the two days): up to 32 points;
  - e.g.: connection from 8:55 to 11:20, two rates: from 00:00 to 9:00 0.125 per minute, from 9:00 to 24:00 0.250 per minute
  - e.g.: connection from 22:59 to 1:12, same two rates as previous line
  - e.g.: connection from 8:55 to 11:20, two rates: from 18:00 to 9:00 0.125 per minute, from 9:00 to 17:00 0.250 per minute
  - e.g.: connection from 22:59 to 1:12, same two rates as previous line

HINT: Internally transform all times in minutes, i.e. time 01:12  $\rightarrow$  60+12 = 72 minutes; then work with minutes, also to check that a connection partially or totally belongs to a window or another.

HINT2: especially for solving Items 2-4, draw and design carefully your algorithm, before starting with the production of the code.

It is not required to provide the optimal (shortest, most efficient, fastest, ...) solution, but a working and clear solution.

It is required to write at class time a short and clear explanation of the algorithm used.

It is required to write at class time to write significant comments to the instructions.

The input-output part is not necessary in the class developed solution, but its implementation is mandatory to be discussed at oral exam.

***Please use carbon copy and retain one copy for home implementation and debug. Please provide your classroom submitted solution with several explanatory and significant comments. When coming to oral discussion, please mark on your "classroom" copy all modifications. Please also provide an error-free and running release of the solution, as well as with its printed list of instructions. Please consider that the above are necessary but not sufficient requirements to success the exam, since the final evaluation will be based on a number of parameters.***

***FAILURE TO ACCOMPLISH ALL PREVIOUS NECESSARY REQUIREMENTS WILL CAUSE NO-QUESTION-ASKED AND IMMEDIATE REJECTION.***