

25 January 2016 -- Computer Architectures -- part 1/2

Matr, Last Name, First Name

The website <http://www.ratemyprofessors.com>, self defines itself as being “the largest online destination for professor ratings in USA, UK and Canada”. The practice of providing students with tools to evaluate their professors is currently rather common at highly reputed universities and institutions. The Fabulous University of No-Nerds (FUN2) uses a polling system where students rate their professors on 4 criteria (C=clarity, A=availability, S=students involvement capabilities, E=efficiency of teaching) using scores from 1 to 6 (a zero denotes a no-rating provided by a student for that criterion). It is well known that each course at FUN2 accounts from a minimum of 11 students up to 42 max and that the number of rated professors is between 2 and 15 and that each professor teaches only one course. It is requested to write a program in 8086 that, upon receipt of a global “university” array of N EQU 630 elements with all scores for all professors, elaborates some statistics. The array SCORES is sorted by professor’s code (from 1 to 15) and the format of one entry of scores array is:

pppp ccc aaa sss eee

with pppp=professor’s code, from 1 to 15 (0000 means that the entry is not valid), and ccc, aaa, sss, eee are the scores for C, A, S, E; please note that 0 indicates that the student did not provide a rating for that criterion. Please also observe that all “non valid entries” (if any) are in the “lowest” part of the array SCORES DW N DUP(?)

For each professor it is necessary to compute the average evaluation for each criterion based on the scores received, i.e. AC, AA, AS, and AE. For each professor it is also necessary to compute the overall average evaluation denoted as Professor’s Score (PS), based on all the scores received in all criteria.

All averages have to be computed on 3 integer and 5 fractional bits (the extra bits can be easily neglected by truncation), i.e. XXX.YYYYY, where all stored within a single byte.

NOTES and HINTS (observe that)

- The array is sorted, i.e. all evaluations on professor 2 follow all evaluations of professor 1 and so on.
- For each valid line there is at least one valid criterion rating.
- Each professor has at least one valid evaluation for each single criterion.
- Right after the last evaluation row of professor 15 the rest of the array is filled with non-valid entries, which are therefore concentrated (if any) at the higher row indexes of the array SCORES.
- Maximum points collected for each criterion is $42 * 6 = 252$ i.e., fitting one byte.
- **To compute each criterion average value, it is necessary to implement a division on an operand on 16 bits divided by another operand on 8 bits; it is therefore necessary to extend the 8 bits storing the accumulated number of points (which is an integer value) to 16 bits in a suitable way. It is recommended to study and properly implement this extension by looking at the format of the target result (3 integer and 5 fractional bits).**
- Observe that, in general, PS is NOT the average of AC, AA, AS and AE, as there could be students who could have decided to rate only on some, but not all, criteria.
- For the computation of PS, students must determine, by looking at the worst cases, the sizes of variables involved in the computation and to do the proper “adjustments” before executing the division operation.

An Example (with 3 professors only)

INPUT				
Prof. code	C	A	S	E
1	5	0	5	4
1	6	6	5	6
1	5	5	4	5
1	5	0	4	0
2	2	1	1	2
2	1	1	1	0
3	2	3	2	2
3	3	4	2	3
3	2	4	3	3
3	2	2	1	2
0	-	-	-	-
0	-	-	-	-

Computations (N_C =number of C votes, etc)								
code	$\sum C$	$\sum A$	$\sum S$	$\sum E$	N_C	N_A	N_S	N_E
1	21	11	18	15	4	2	4	3
2	3	2	2	2	2	2	2	1
3	9	13	8	10	4	4	4	4

- denotes a don't care

RESULTS ($AC = \sum C / N_C$, etc)				
AC	AA	AS	AE	PS
5.25	5.5	4.5	5	65/13=5
1.5	1	1	2	9/7=1.28...
2.25	3.25	2	2.5	40/16=2.5