17 February 2016 -- Computer Architectures -- part 1/2

Matr, Last Name, First Name ....

"*The future is written in the stars... and the cards*" This is the motto of the University of Grass W-egas where scores of exams are computed with the help of playing cards plus a small set of choices (when available) operated by Students. Before exams start the professor scrambles 4 decks of cards (each one with four seeds and values from ace to 10, plus Jack, Queen and King). Students are then, individually, asked 3 technical questions related to exam's topics. Each correctly answered question provides the student with one **token** corresponding to one **opportunity** to draw one card, starting from the (current) top of the deck. Drawn cards will determine exam's grade according to some rules.

BASIC RULES: *After all questions have been asked & responded, the Student who awarded some* ***tokens****, draws the first card (from the top of the deck), reads its value and decides whether to proceed further or not in case he/she has additional* ***tokens****. The final score of the exam is obtained by adding the values of drawn cards to a* ***"base score" = 6****.*

If the final score is between 18 and 30, then the exam is passed and a grade equal to the achieved score is recorded for that exam. If the final score is outside the range [18, 30] then the exam is failed and no grade is recorded. Please observe that each student can draw as many cards as the number of **tokens** she/he has in the hands, but no more; i.e. if a student has 2 **tokens**, she/he can decide to draw 1 or 2 cards, but not 3. Values of cards correspond to the number printed on the card, with ace=1, two=2 and so on, including Jack=11, Queen=12, King=13.

- For First year's Students there are no additional rules besides the BASIC RULES stated above.
- For Second year's Students BASIC RULES hold and students are give extra possibilities, according to the ADDENDUM RULE described further.
- For Third year's Students BASIC+ADDENDUM RULES are integrated with the possibilities offered by the EXTRA RULE described further.
- For Fourth and Fifth years' Students (corresponding to Master) BASIC+ADDENDUM+EXTRA RULES hold, plus the SUPER RULE described further.

ADDENDUM RULE: *After a Student has drawn all cards she/he decided-to/could draw, she/he could decide that all characters (i.e. Jack, Queen, King) bring 10 and not the corresponding points.*

EXTRA RULE: *After a Student has drawn all cards she/he decided-to/could draw, she/he could decide that all ace cards bring 14 points and not 1. (This choice is independent of the choice taken according to ADDENDUM RULES).*

SUPER RULES: *After a Student has drawn all cards and decided which opportunity to use according to ADDENDUM and EXTRA RULES, she/he could decide whether the points achieved by drawn cards are* ***added to the "base score" = 6*** *or are* ***subtracted by a "ceil score" = 52****.*

In addition there is also a **SUPERBONUS RULE**, which applies only to some Students: *Usually, at the University of Grass W-egas, professors do not award 30 cum laude, but for 2016 only, to celebrate his 25th anniversary of teaching, professor OMP awards 30 cum laude to all Students (with any seniority), drawing three "7" cards.*

It is requested to write a 8086 assembly program to implement the grading, strictly according to Student's choices, i.e. without offering the different values in case different choices are taken, but simply "obeying to Student's choices". The program receives in input the array DECKS of 4*52= 208 cells, holding cards values in the following format:

xx ss vvvv where ss=suit (spades, hearths, diamonds, clubs), vvvv is card's value from 1 to 13 and xx are not used.

For each Student (up to 30) with at least one token, the program asks:

- Number of tokens (from 1 to 3);
- Seniority (i.e. attendance year) of the Student (i.e., a number from 1 to 5);
- If the Student belongs to Professor's OMP Classes.

Then, the program takes the card at the current top of deck and displays its value and seed. If the Student has additional tokens she/he is then asked to draw or not draw more cards. At the end of drawing, according to Student's seniority, the corresponding rules are implemented, i.e. for Students of first year BASIC RULES are immediately implemented, while for more senior Students they are asked about the actions to be taken according to ADDENDUM, EXTRA and SUPER RULES. For Students of Professor's OMP Classes, prior implementing the BASIC RULES, the SUPERBONUS RULE is checked & handled. After processing one Student the program continues with the next.

17 February 2016 -- Computer Architectures –- part 1/2

Matr, Last Name, First Name ....

A few Examples:

- Student with 3 tokens, seniority=1, not from Professor's OMP Classes:
  First draw: K clubs. Student decides to draw another card; second draw: 4 spades. Student decides to stop;
  Final grade = 13+4+6 (i.e. the base value) = 23.
- Student with 3 tokens, seniority=2, not from Professor's OMP Classes:
  First draw: Q clubs. Student decides to draw another card; second draw: 2 hearts. Student decides draw another card; third draw: K hearts. Student is asked to re-size (if any) all J,Q,K to 10 points; student agrees to.
  Final grade = 10+2+10+6 (i.e. the base value) = 28 (otherwise, score would have been 12+2+13+6 = rejected)
- Student with 2 tokens, seniority=3, from Professor's OMP Classes:
  First draw: J diamonds. Student decides to draw another card; second draw: 1 spades (i.e. ace of spades). Student cannot draw more cards as she/he has no more tokens. Student is asked to re-size (if any) all J,Q,K to 10 points; student agrees to. Student is asked to re-size (if any) all ace(s) to 14 points; student agrees to. Final grade = 10+14+6 (i.e. the base value) = 30 (otherwise, score would have been 11+1+6 = 18)
- Student with 2 tokens, seniority=4, from Professor's OMP Classes:
  First draw: 1 (i.e. ace) diamonds. Student decides to draw another card; second draw: 8 spades. Student cannot draw more cards as she/he has no more tokens. Student is asked to re-size (if any) all J,Q,K to 10 points; independently of student's choice there will be no effect as there are no J/Q/K. Student is asked to re-size (if any) all ace(s) to 14 points; student agrees to. Student is asked to add to base score=6 or to subtract from ceil score=52; student decides to subtract. Final grade = 52-(14+8) = 30

Tasks to be implemented and corresponding point (only fully completed items will be considered to award points)

- Item 1 (MANDATORY): write a running program asking Students the 3 questions, dealing card drawing, implementing BASIC RULES and computing the final score accordingly. POINTS → 24
- Item 2: in addition to Item 1, write a running program implementing ADDENDUM RULE and computing the final score accordingly. POINTS → +4
- Item 3: in addition to Item 1, write a running program implementing EXTRA RULE and computing the final score accordingly. POINTS → +4
- Item 4: in addition to Item 1, write a running program implementing SUPER RULE and computing the final score accordingly. POINTS → +4
- **Bonus Item**: in addition to Item 1, write a running program implementing "Professor's OMG 30-cum-laude" SUPERBONUS rule and computing the final score accordingly. POINTS → +2

Please consider that a maximum of 33 points can be accounted here; larger values will be "cut off" to 33.

HINTS (observe that)

- The program should NOT find and/or implement the best choices, but only "obey" to Student's choices;
- It is advised to design the program as a collection of modules, each one implementing the different rules.

REQUIREMENTS (SHARP)

- It is not required to provide the optimal (shortest, most efficient, fastest) solution, but a working and clear one.
- 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 significant comments to the instructions.
- Input-output is not necessary in class-developed solution, but its implementation is mandatory for the oral exam.
- Minimum score to "pass" this part is 15 (to be averaged with second part and to yield a value at least 18)

REQUIREMENTS ON THE I/O PART TO BE DONE AT HOME

- The databases (if any, i.e. not necessary in case) have to be defined and initialized inside the code
- All inputs and outputs should be in readable ASCII form (no binary is permitted).

Please use carbon copy ONLY (NO PICTURES ARE ALLOWED) and retain one copy for home implementation and debug. At the end of the exam please give to professors all the sheets of your solution. Missing or late sheet will not be evaluated. Please provide your classroom submitted solution with several explanatory and significant comments. Please remember that only what has been developed at class time can and will be evaluated at oral time and that it is necessary to write the instructions of the program and not just the description of the algorithm. When coming to oral discussion, please clearly mark in red 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 THE ABOVE NECESSARY REQUIREMENTS WILL CAUSE NO-QUESTION-ASKED AND IMMEDIATE REJECTION.