A2013 Final Exam LO21 and LO27 Wednesday, January 15, 2013

Terms:

- Duration: 2 hours
- Documents, calculator and phone are not authorized
- The rate is indicative (±1)
- One sheet per exercise

Exercise 1 (11 points) – Lists

An on-demand transport service has a list of vehicles that it makes available to its customers. Each vehicle is characterized by its identifier (a nonzero integer), its capacity (the maximal number of people that it can carry), and its availability (a Boolean). The list of vehicles is sorted in ascending order on the availability of vehicles.

1) Given a transport request (an integer identifying the demand and an integer indicating the number of persons to be carried), write the recursive algorithm of the subroutine ***Disp*** which gives the identifier of the first vehicle available for meeting this demand. (3 points)

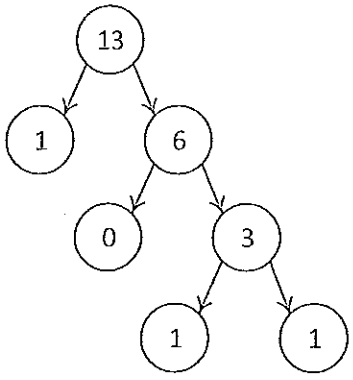
2) Given a list of requests sorted in ascending order of the number of persons to be transported and a list of vehicles, write the algorithm of the iterative subroutine ***Sat*** that builds the list of demand's identifiers whose demand can be met. (3 points)

3) Give the recursive version ***rSat*** of the previous algorithm ***Sat***. (3 points)

4) Representing the list of vehicles by a simple linked list, write in C language the complete declaration of the type "*ListOfVehicle*" and the translation of the subroutine ***Disp***. (2 points)

Exercise 2 (9 points) – Binary Trees

We consider the representation as a binary tree of the successive steps of the binary-decimal conversion of an integer. Here the binary tree t corresponding to the integer 13.
$(13)_{10} = (1101)_2$



1) Provide the recursive algorithm of the subroutine ***binTree*** which builds the binary tree for a given nonnegative integer. (3 points)

2) Provide the recursive algorithm of the subroutine ***bin*** which gives the binary representation of the associated integer in form of a list of integer from a binary tree. E.g. bin(t) = $(1101)_2$ (3 points)

3) Provide the recursive algorithm of the subroutine ***sumDiv*** which gives the sum of the dividend and all quotients from a binary tree. E.g. sumDiv(t) = 13 +6 +3 +1 = 23 (3 points)