Esame di ammissione al dottorato in Informatica anno 2009

1. [**Exercises**] The candidate is required to choose and solve two excercises out of the four below:
   (a) Consider the following Java language fragment:

Exp	::=	Num | Bool | Ide | **(Exp)** | **Exp Op Exp** | **!Exp**
Op	::=	+ | - | * | % | / | == | != | < | <= | > | >= | && | ||
Com	::=	Ide = **Exp ;** | **if (Exp) Com else Com** | **while (Exp) Com** | **Block**
Block	::=	**{ StList }**
StList	::=	**Stat** | **Stat StList**
Stat	::=	**Com** | **Decl**
Decl	::=	**int** Ide **;** | **boolean** Ide **;**

   where Num, Bool and Ide have the obvious syntax. For static semantics analysis purposes, define a map $\mathcal{W} : \mathbf{Com} \to \{\mathbf{tt}, \mathbf{ff}\}$ such that, for each command, the value **tt** is returned if and only if every identifier has been declared in the command before being used.
   Make sure to take into account the nested structures of blocks and focus on the problem of declaration presence / absence only, without paying attention to type mismatch.
   (b) Let a be an array of n different integers such that there exists an index j with $0 \leq j < n$ that satisfies all of the following:
   - the elements in the segment $a[0, j]$ are listed in decreasing order;
   - the elements in the segment $a[j, n-1]$ are listed in increasing order;

   Given the problem of determining the index j, discuss its complexity and define a possibly optimal algorithm for its solution.
   (c) The algorithm (Dekker) below is the first known programmed solution for the critical section problem with two processes involved. Let P_0 and P_1 share the variables:

```
boolean flag[2]; /*initially false*/
int turn;
```

   The skeleton of P_i (for $i \in \{0, 1\}$) is below (where j denotes the index of the other process):

```
do {
    flag[i] = true;
    while (flag[j]) { // assume j==1-i
        if (turn == j) {
            flag[i] = false;
            while (turn==j)
                ; // do nothing
            flag[i]=true;
        }
    }
    // [critical section]
    turn = j;
    flag[i]=false;
    // [non critical section]
} while (true);
```

   Show that the algorithm satisfies the three conditions (namely, mutual exclusion, progress and bounded waiting) required to any solution of the critical section problem.
   (d) Consider the problem related to the data management for the schools in a region of Italy. For each school, it is known the municipal it pertains to, the name of the school, the overall number of students and the kinds of classes offered (elementari, medie, liceo, istituto tecnico,...). For each kind of class, it is known the duration and the age at which students start that class (in years). A municipal is allowed to have schools only if it is classified as main (e.g. depending on the number of inhabitants). The other municipals are allowed to use schools in a given main municipal of reference. For each municipal, it is known the number of inhabitants, the adjacent municipals and the main municipal of reference (if needed).
   Draw the object scheme for the database, then transform it in the corresponding logic scheme of a relational database and give its graphical representation. For each relation in the scheme, make explicit all attribute names, the primary key and external keys.

2. [**Dissertation**] Take one of the four areas listed below, point out an open probolem within that area and describe its state of the art and the future possible developments:
   - Peer-to-peer networks.
   - The comparison between $\mathcal{P}$ and $\mathcal{NP}$ complexity classes.
   - Knoweldge discovery systems.
   - Web services and their standardization process.