Esame di ammissione al Dottorato di Ricerca in Informatica, anno2011

1. Il candidato svolga due esercizi a scelta tra i quattro seguenti

a) Assumiamo come dato un insieme A = {a, a', …}di azioni. Un sistema di transizione su A è una struttura (Q, T) dove Q = {q, q', …} è l'insieme degli stati e $T \subseteq Q \times A \times Q$ è la relazione di transizione. Un sistema di transizione deve essere pensato come un automa (non necessariamente a stati finiti) senza stato iniziale e senza stati di accettazione.

Introduciamo una semplice logica modale per descrivere proprietà relative all'evoluzione dello sistema. La sintassi delle formule logiche P risulta:

$$P ::= tt \mid ff \mid P \wedge P \mid P \vee P \mid \langle a\rangle P \mid [a]P$$

Di seguito viene presentata la relazione di soddisfacibilità, ovvero quando uno stato q soddisfa una formula P ($q \models P$) o quando non la soddisfa ($q \not\models P$):

$q \models tt$ $\qquad$ $q \not\models ff$

$q \models P1 \wedge P2 \Longleftrightarrow q \models P1$ e $q \models P2$ $\qquad$ $q \models P1 \vee P2 \Longleftrightarrow q \models P1$ o $q \models P2$

$q \models \langle a\rangle P \Longleftrightarrow \exists\, q' \in Q$, tale che$(q, a, q') \in T$ vale che $q' \models P$

$q \models [a]P \Longleftrightarrow \forall\, q' \in Q$, tale che$(q, a, q') \in T$ vale che $q' \models P$

Due stati q e q' sono logicamente indistinguibili (e scriviamo $q \simeq q'$) se e solo se soddisfano le stesse formule logiche

i) Dimostrare che $\simeq$ è una relazione di equivalenza.

ii) Siano M1 = (Q1, q, F1, T1) e M2 = (Q2, q', F2, T2) due automi a stati finiti sullo stesso alfabeto. Sotto quali condizioni l'equivalenza logica ($q \simeq q'$) implica che i due automi riconoscono lo stesso linguaggio. Tramite un esempio mostrare che la tradizionale equivalenza tra automi (due automi sono equivalenti se riconoscono lo stesso linguaggio) non implica l'equivalenza logica.

b) Si descriva la tecnica di divide-et-impera per lo sviluppo di algoritmi efficienti, fornendo uno specifico esempio di problema per il quale tale tecnica consente di trovare la soluzione in tempo ottimale.

c) Si consideri una computazione a livello di processi, consistente di un processo Master e dei processi identici $Worker_0$, …, $Worker_{n-1}$, capaci di eseguire una funzione F da interi a interi. Il tempo di esecuzione della funzione F è fortemente variabile a seconda del valore del parametro d'ingresso.

Il processo Master riceve una sequenza, eventualmente illimitata, di valori interi. Il suo compito è di schedulare i valori della sequenza d'ingresso nei confronti dei worker. La strategia di scheduling deve assicurare un buon bilanciamento del carico dei worker a fronte dei tempi di esecuzione variabili della funzione F.

Esprimere la computazione con un formalismo a scambio di messaggi a scelta del candidato. Di tale formalismo si dia prima una breve caratterizzazione.

La soluzione non deve far uso di ipotesi sul supporto a tempo di esecuzione del livello dei processi.

d) Si definisca nel modello relazionale uno schema di base di dati che rappresenta un grafo (non diretto) G composto da un insieme di nodi V e di archi E, e si definiscano le seguenti interrogazioni, in riferimento allo schema introdotto, utilizzando l'algebra relazionale oppure la sintassi SQL:

1) *degree*(x, k): l'interrogazione deve restituire una relazione *degree* che associa ad ogni nodo x il numero k degli archi incidenti su x;

2) *degree-distribution*(k, n): l'interrogazione deve restituire una relazione che associa ad ogni valore distinto $k>0$ del grado, il numero n dei nodi che possiedono quel grado;

3) *common-neighbors*(x, y, m) : l'interrogazione deve restituire una relazione che associa ad ogni coppia di nodi x, y il numero m dei nodi adiacenti comuni (raggiungibili sia da x che da y con un solo arco); in formula

$$m = \left|\Gamma(x) \cap \Gamma(y)\right|$$

dove, per ogni nodo x , $\Gamma(x) = \{z \in V : (x,z) \in E\}$ è l'insieme dei vicini di x.

4) *adamic-adar*(x, y, w) : l'interrogazione deve restituire una relazione che associa ad ogni coppia di nodi x, y il valore w ottenuto sommando, per ciascun nodo z adiacente sia ad x che ad y, l'inverso del logaritmo del grado di z; in formula, dati due nodi ⍰, ⍰, il valore ⍰ della misura di Adamic-Adar è definito da:

$$w = \sum_{z \in \Gamma(x) \cap \Gamma(y)} \frac{1}{\log\left|\Gamma(z)\right|}$$

2. Nell'ambito di una delle aree di ricerca sotto elencate, individuare un problema di ricerca e discuterlo:

a) basi di dati

b) complessità computazionale

c) linguaggi di programmazione

d) sistemi paralleli e distribuiti