

Cognome ..... Nome ..... Matricola .....

## Sistemi Informativi

18-07-2017

Sbarramento							
1 (4pt)	2 (4pt)	3 (6pt)	4 (5pt)	5 (4pt)	6 (4pt)	7 (6pt)	Somma (33pt)

Il compito ha la durata di 3 ore, per la compilazione attenersi alle seguenti istruzioni:

- Nome e cognome devono essere scritti in **STAMPATELLO**.
- É sola responsabilità dello studente scrivere in modo **LEGGIBILE**.
- Solo le soluzioni scritte all'interno dei riquadri verranno corrette.
- Non sono ammessi appunti o altro materiale, a chiunque copia verrà ritirato l'esame.
- Affinché l'esame venga corretto e quindi valutato, negli esercizi marchiat **con \*\*** si dovrà raggiungere almeno 6 punti come somma totale.
- L'esame si ritiene superato se si raggiunge il punteggio di 18. La lode si ottiene con punteggio  $\geq 31$ .
- Qualsiasi soluzione ragionevole è accettata ma verranno premiate soluzioni complete in termini di sintassi e valutata positivamente la padronanza dei concetti presentati durante il corso.
- Sono ammesse le seguenti abbreviazioni nell'XML schema e nel DTD:
  - `<e ...> = <!ELEMENT ... >` e `<xs:element ... >`
  - `<a ...> = <!ATTLIST ... >` e `<xs:attribute ... >`
  - `<ct ...> = <xs:complexType ... >`
  - `<st ...> = <xs:simpleType ... >`

Si riportano alcuni commenti che aiutano nella risoluzione,  
ma **NON** costituiscono così come sono una soluzione  
**COMPLETA.**

Potrebbero contenere errori di battitura

Il materiale per risolvere i restanti esercizi è nelle slide e nelle dispense.

**Esercizio 1 (4 punti).** Dato lo Schedule:

S: r1(A); r1(B); r2(B); r3(B); r2(B); w1(C); w2(C); r3(C); r2(C); w3(C); w3(A); w3(D)  
**1pt)** \*\* Dimostrare se S è **conflict serializzabile**.

Qualsiasi operazione di tipo scrittura genera conflitti con qualsiasi altra operazione sulla stessa risorsa

Due operazioni di read, non generano conflitti. Quindi: r2(B);r3(B) non genera conflitti.

r1(A);w3(A) Sì.

w1(C);w2(C) Sì.

I conflitti nel grafo sono: T1→;T2, T1→;T3, T2→;T3.

Il grafo delle precedenze crea un anello, ma NON un ciclo. Il grafo orientato è aciclico. Lo schedule è conflict equivalente allo schedule seriale T1;T2;T3.

**1pt)** Aggiornare lo schedule inserendo le operazioni per shared, exclusive e upgradable lock.

Si mette  $sl_i(X)$  davanti alla prima read di  $T_i$  sulla risorsa  $X$ ,  $xl_i(X)$  davanti alla prima write non preceduta da read.

Quando una transazione finisce chiama unlock  $u_i(X)$  per ogni risorsa. Si usa updatable lock  $ul_i(X)$  se  $T_i$  prima legge  $X$  e poi scrive anche su  $X$ .

$sl_1(A); r_1(A); sl_1(B); r_1(B)...ul_3(C); r_3(C); r_2(C); u_2(B); u_2(C); xl_3(C); w_3(C)...$

**2pt)** Usare shared, exclusive, e upgradable locks. Descrivere cosa fa ogni parte di un scheduler 2PL con le prime 7 operazioni di S

Sia dato il seguente XML contenuto nel file "doc.xml":

```
<?xml version="1.0" encoding="ISO8859-1" ?>
<Universita>
  <Dipartimento id="d1"> <!-- almeno 1 -->
    <Indirizzo tipo="via" nome="Verona" numero="12" /> <!-- tipo: via o piazza -->
    <Corsi>
      <Cor codice="c1" /> <!-- almeno 1 -->
    </Corsi>
  </Dipartimento>
  <Dipartimento id="d2">
    <Indirizzo tipo="piazza" nome="Venezia" numero="9" />
    <Corsi>
      <Cor codice="c1" /> <!-- si riferisce a 'Corso' -->
      <Cor codice="c2" />
      <Cor codice="c3" />
    </Corsi>
  </Dipartimento>
  <Corso codice="c1" anno="2013">
    <Descrizione>Sistemi informativi</Descrizione>
    <Crediti>6</Crediti> <!-- solo 6,8 o 12 -->
    <Esercitatore cf="ADB122" /> <!-- si riferisce a 'Professore' -->
  </Corso>
  <Corso codice="c2" anno="2009">
    <Descrizione>Database avanzato</Descrizione>
    <Crediti>8</Crediti>
  </Corso>
  <Corso codice="c3" anno="2012">
    <Descrizione>Programazione</Descrizione>
    <Crediti>12</Crediti>
  </Corso>
  <Professore cf="ADB122" rettore="si"> <!-- se non specificato = no -->
    <Nome>Pippo</Nome>
    <Cognome>Rossi</Cognome>
    <Stipendio>1200</Stipendio>
  </Professore>
  <Professore cf="ADB123" >
    <Nome>Gianni</Nome>
    <Cognome>Verdi</Cognome>
    <Stipendio>1150</Stipendio>
  </Professore>
  <Professore cf="ADB124" >
    <Nome>Antonio</Nome>
    <Cognome>Bianchi</Cognome>
    <Stipendio>1300</Stipendio>
  </Professore>
</Universita>
```

**Esercizio 2 (4 punti).** Rispondere alle seguenti domande

**2pt)** Scrivere un possibile schema DTD valido (escludendo il contenuto di <Professore>)



**2pt)** Scrivere il frammento di un possibile XML schema degli elementi <Corso> includendo le chiavi esportate





**Esercizio 3 (6 punti).** Si consideri il contenuto XML precedente.

**3pt)** Scrivere l'interrogazione XQuery per restituire il dipartimento (compreso tutto il tag) che ha piú corsi. `<Dipartimento>...</Dipartimento>`

**3pt)** Scrivere il risultato dell'esecuzione della seguente query (si assuma che `$doc` sia ben definito).

```
let $resp := (  
  for $c in $doc//Corso, $d in $doc//Dipartimento  
  where data($c/@codice) > 'c2'  
  order by data($c/Crediti)  
  return <i>{$c}</i>  
)  
return <list>{$resp}</list>
```



**Esercizio 4 (5 punti).** Siano dati due sistemi per la gestione di magazzini, ognuno con un proprio database:

DB1:	DB2
Prodotto(Codice, Nome, Descrizione, IDCat)	Item(Code, Name, Description, Category, Quantity)
Categoria(ID, Nome, Descrizione)	Category(ID, Name, Producer)
Variante(CodiceProd, IDVariante, Quantita)	

**2pt) \*\*** Per integrare i dati delle due tabelle, scrivere due schema globali, uno per ‘Prodotto/Item’ e uno per ‘Categoria/Category’

*Si consideri e si descriva con attenzione come gestire la quantità di prodotto supponendo di voler mantenere all’interno dello schema globale le distinzioni di DB1 riguardo le ‘Varianti’.*

`Prod(Cod, Nome, Desc, IDCat, IDVar, QTY)`

`Cat(IDCat, Nome, Desc, Producer)`

Si assume per DB2 un codice variante `IDVar` che non sia presente in DB1 Varianti. Le quantità per DB2 e per DB1 rimangono invariate, se si vuole sapere la quantità complessiva di un prodotto da DB1, si utilizza il `GROUP BY` e `SUM`.

**3pt)** Supponendo di avere i seguenti adornments a disposizione per DB1

`Prodotto (bffu, ubfu, uufc[c1...c999]))`    `Categoria(bfu, ufu)`

Discutere un piano per eseguire la query:

```
SELECT p.nome, p.descrizione FROM Prodotto p JOIN Categoria c ON p.IDCat=c.ID
WHERE c.Nome = "antani";
```



**Esercizio 5 (4 punti).** Considerare le seguenti 3 matrici.

**2pt)**  $\star\star$  Motivare se e quali di esse siano valide matrici di transizione per il calcolo del Page Rank.

	x1	x2	x3	x4	x5
x1	1				
x2		1			
x3			1		
x4				1	
x5					1

	x1	x2	x3	x4	x5
x1	1/3	1/3			
x2		1/3	1/3		
x3	1/3	1/3	1/3		1/3
x4	1/3			1/3	1/3
x5				1/2	1/2

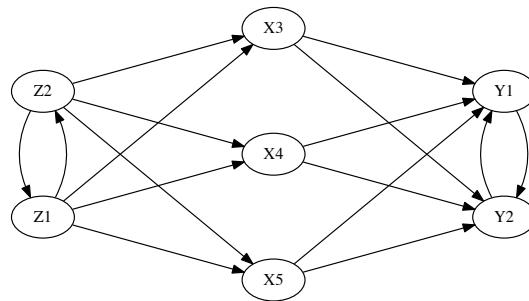
	x1	x2	x3	x4	x5
x1	1/3			1/2	
x2		1/2			1/3
x3		1/2	1/2		
x4	1/3		1/2	1/2	1/3
x5	1/3				1/3

Le matrici devono essere quadrate (per ogni nodo, transizione ad ogni altro nodo).

Le colonne devono sommare a 1.

Ogni colonna rappresenta la *probabilità di transizione da un nodo a tutti gli altri*.

Non è necessario che le colonne contengano numeri uguali, ci sono modelli di PageRank in cui archi diversi hanno maggiore probabilità di transizione.



Teleport Set =  $\{X3, X4, X5\}$

**2pt)** Dato il grafo in figura, calcolare pagerank con  $\beta = 0.8$  utilizzando le prime 2 iterazioni del metodo con relaxation

$$z1_i = \frac{8}{10} \cdot \frac{1}{4} \cdot z2_{i-1}$$

$$z2_i = \frac{8}{10} \cdot \frac{1}{4} \cdot z1_{i-1}$$

$$x3_i = \frac{8}{10} \left( \frac{1}{4} \cdot z1_{i-1} + \frac{1}{4} \cdot z2_{i-1} \right) + \frac{2}{10} \cdot \frac{1}{3}$$

$$x4_i = \frac{8}{10} \left( \frac{1}{4} \cdot z1_{i-1} + \frac{1}{4} \cdot z2_{i-1} \right) + \frac{2}{10} \cdot \frac{1}{3}$$

$$x5_i = \frac{8}{10} \left( \frac{1}{4} \cdot z1_{i-1} + \frac{1}{4} \cdot z2_{i-1} \right) + \frac{2}{10} \cdot \frac{1}{3}$$

$$y1_i = \frac{8}{10} \left( \frac{1}{2} \cdot x3_{i-1} + \frac{1}{2} \cdot x3_{i-1} + \frac{1}{2} \cdot x5_{i-1} + y2_{i-1} \right)$$

$$y2_i = \frac{8}{10} \left( \frac{1}{2} \cdot x3_{i-1} + \frac{1}{2} \cdot x3_{i-1} + \frac{1}{2} \cdot x5_{i-1} + y1_{i-1} \right)$$

Prima iterazione  $i = 1$ : A destra si sostituisce  $z1_0 = z2_0 = x3_0 = x4_0 = x5_0 = y1_0 = y2_0 = \frac{1}{7}$  e si calcola  $z1_1; z2_1; x3_1; x4_1; x5_1; y1_1; y2_1$ .

Seconda iterazione  $i = 2$ , si usano i valori di  $z1_1; z2_1; x3_1; x4_1; x5_1; y1_1; y2_1$ , per calcolare  $z1_2; z2_2; x3_2; x4_2; x5_2; y1_2; y2_2$ .



**Esercizio 6 (4 punti).** Dato un grafo bipartito formato dal set di nodi :  $\{a_0, a_1, \dots, a_n, b_0, b_1, \dots, b_n\}$  dove esiste un arco  $(a_i, b_j)$  se e solo se rispetta la proprietà  $P : i = \lceil j/2 \rceil \vee i = 2 + \lceil j/2 \rceil$

**2pt) \*\*** Si trovi il matching perfetto del grafo tra nodi  $(a_i, b_j)$  per  $n = 6$

Si usano i valori possibili di  $j = \{0, 1, 2, 3, 4, 5, 6\}$ .

$$j = 0; i = \lceil 0/2 \rceil \vee i = 2 + \lceil 0/2 \rceil \rightarrow i = 0, i = 2$$

$$j = 1; i = \lceil 1/2 \rceil \vee i = 2 + \lceil 1/2 \rceil \rightarrow i = 1, i = 3$$

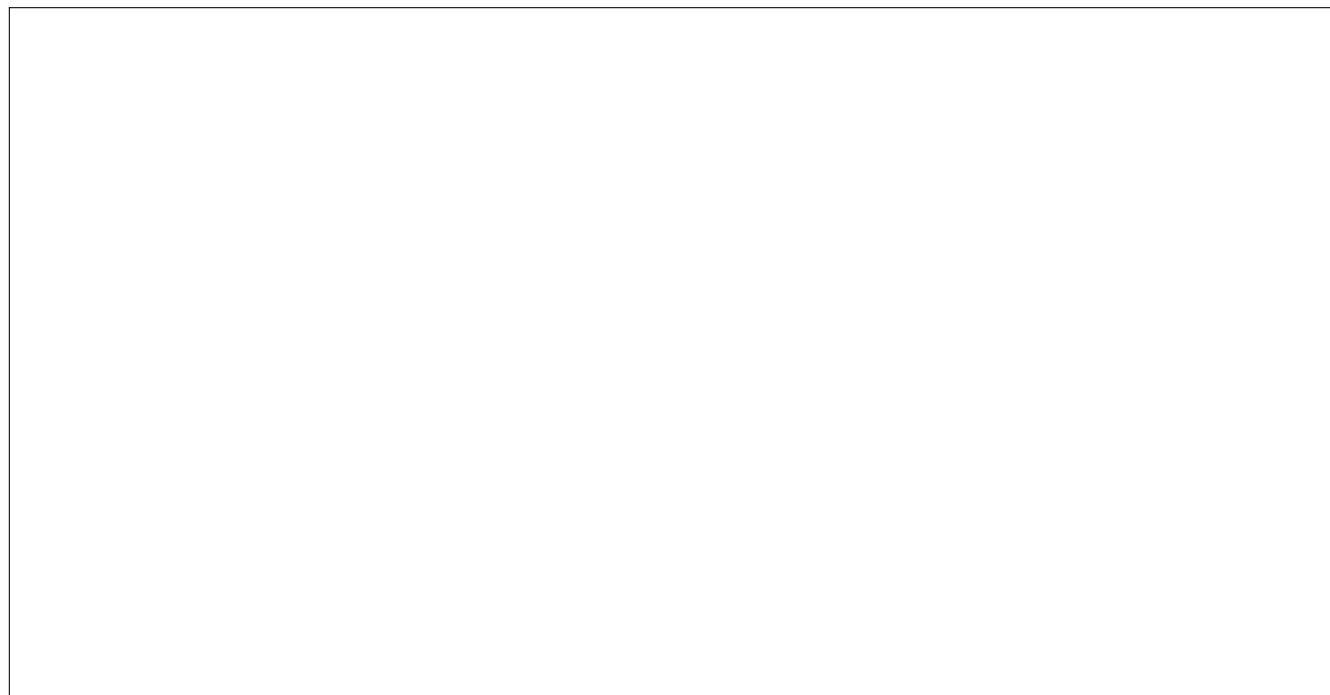
$$j = 2; i = \lceil 2/2 \rceil \vee i = 2 + \lceil 2/2 \rceil \rightarrow i = 1, i = 3$$

$$j = 3; i = \lceil 3/2 \rceil \vee i = 2 + \lceil 3/2 \rceil \rightarrow i = 2, i = 4$$

...

$a_6$  non trova nessun matching, ma questo è un problema di input. Il matching perfetto si ottiene collegando ognuno di  $a_0, a_1, a_2, a_3, a_4, a_5$  ad un singolo  $b_i$ .

**2pt)** Si trovi il matching dato dall'algoritmo greedy considerando l'ordine lessicografico degli archi, ovvero  $(a_0, b_1)$  viene prima di  $(a_0, b_2)$  e prima di  $(a_1, b_0)$ , per  $n = 6$ .







**Esercizio 7 (6 punti).** Siano dati i seguenti elementi  $Aa, Bb, Cc, Dd, Ee$ , che appaiono nei basket:

$$B1 = \{Aa, Bb\}$$

$$B2 = \{Aa, Ee\}$$

$$B3 = \{Bb, Cc, Dd\}$$

$$B4 = \{Aa, Cc, Dd\}$$

$$B5 = \{Aa, Cc, Dd, Ee\}$$

$$B6 = \{Bb, Cc, Dd, Ee\}$$

$$B7 = \{Bb, Cc, Dd, Ee\}$$

$$B8 = \{Aa, Cc, Dd, Ee\}$$

**2pt)**  $\star\star$  Calcolare la Jaccard Distance tra  $B1, B5$ , e  $B7$ .

La distanza è  $1 - JaccardSimilarity = 1 - \frac{|X \cap Y|}{|X \cup Y|}$ .

**2pt)** Calcolare supporto e confidenza per  $Bb, Cc \rightarrow Dd, Ee$  e per  $Aa, Ee \rightarrow Cc$ .

**2pt)** Calcolare la similarità di Jaccard con min-hash tra  $B1, B5$ , e  $B7$  utilizzando le seguenti permutazioni.

P1:  $Aa, Bb, Cc, Dd, Ee$ ,    P2:  $Cc, Aa, Dd, Bb, Ee$ ,    P3:  $Ee, Cc, Dd, Aa, Bb$