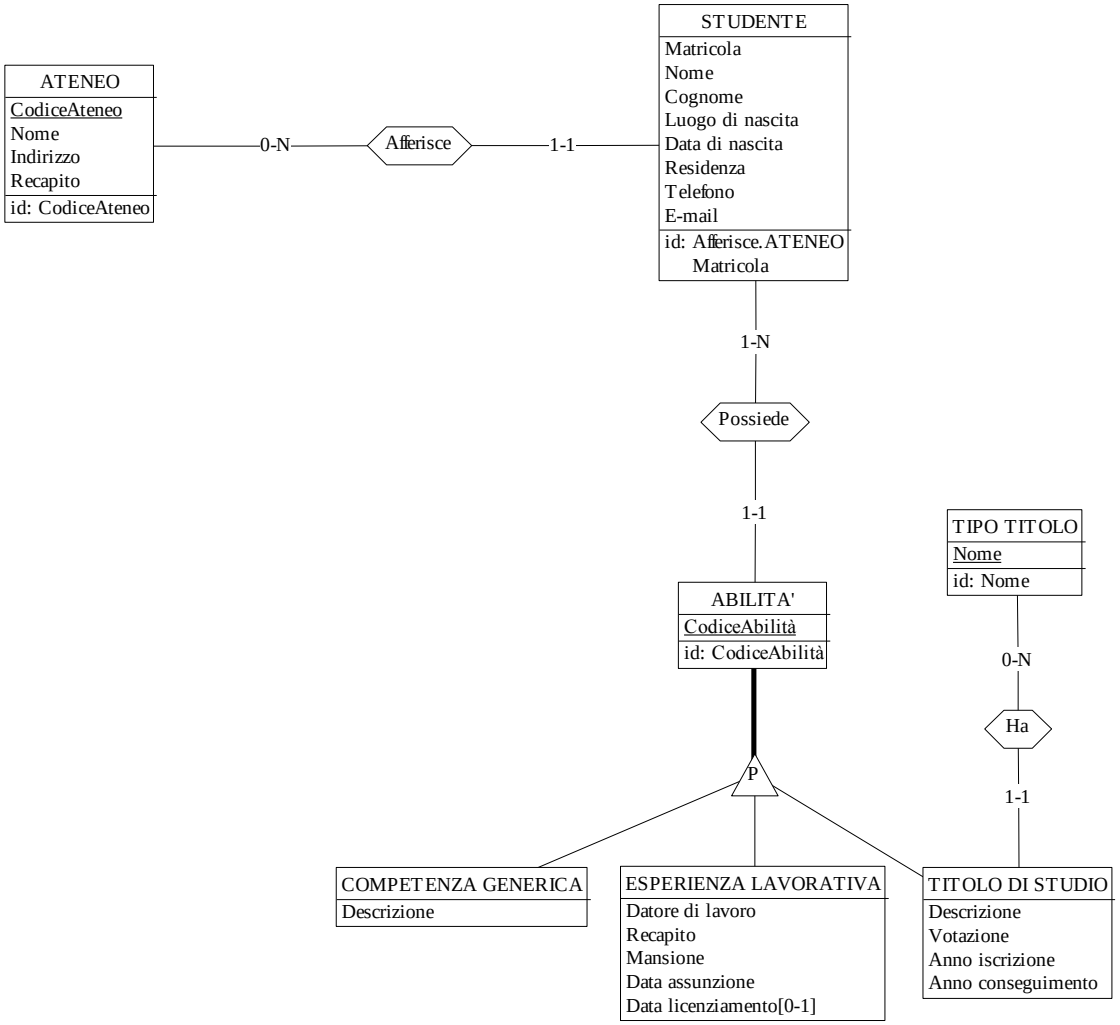
Compito Basi di Dati. Tempo concesso: 2 ore **18 Febbraio 2013**
Nome: _____ Cognome: _____ Matricola: _____

Esercizio 1

Si considerino le seguenti specifiche relative alla realizzazione della base dati a supporto di un applicativo commissionato da un centro di collocamento inter-universitario. Si definisca il relativo ***schema E/R*** *(nella metodologia proposta a lezione). Si evidenzino eventuali* ***vincoli inespressi e attributi derivati****.*

Un centro di collocamento inter-universitario vuole realizzare un software per la gestione del collocamento degli studenti iscritti agli atenei afferenti.
Ogni studente è identificato da una matricola, codice alfanumerico univoco all'interno dell'ateneo di provenienza; dello studente interessano poi nome, cognome, luogo e data di nascita, indirizzo di residenza, recapito telefonico e recapito e-mail.
Di ogni ateneo, identificato da un codice univoco, si vogliono memorizzare nome, indirizzo e recapito di un referente.
Di ogni studente deve essere possibile elaborare in modo automatico un curriculum vitae in formato PDF che comprenda, oltre ai dati anagrafici, una serie di abilità ad esso riconducibili, che possono essere di tre distinte tipologie (titoli di studio, esperienze lavorative, competenze generiche).
Per le competenze generiche interessa memorizzare solo una descrizione testuale. Per ogni esperienza lavorativa interessa invece memorizzare il datore di lavoro, un recapito per la verifica delle referenze, la mansione svolta, la data di assunzione e di licenziamento (ove presente).
Ogni titolo di studio deve essere caratterizzato da una votazione, una descrizione, un anno di iscrizione e un anno di conseguimento. I titoli di studio possono appartenere a cinque tipologie: dottorato di ricerca, master, laurea specialistica, laurea, diploma di maturità.
Ogni abilità associata a uno studente conferisce un punteggio che concorre a formare un punteggio detto *student rank*. In particolare vengono conferiti 0,5 punti per ogni competenza generica, 1 punto per le esperienze lavorative da 3 a 6 mesi, 2 punti per le esperienze lavorative da 6 mesi a 2 anni, 3 punti per le esperienze lavorative oltre i 2 anni. I punteggi relativi ai titoli di studio sono i seguenti: dottorato di ricerca (3), master (1), laurea specialistica (2), laurea (1), diploma di maturità (1).
Il software deve poter ordinare gli studenti in base al ranking e deve mantenere lo storico di tutte le esperienze lavorative e dei titoli di studio conseguiti.

Compito Basi di Dati. Tempo concesso: 2 ore **18 Febbraio 2013 Matricola: _____**



Compito Basi di Dati. Tempo concesso: 2 ore **18 Febbraio 2013 Matricola: _____**

Esercizio 2

- Descrivere l'organizzazione di un B+-tree, evidenziando anche le differenze rispetto alla struttura B-tree.

- È data la query:

  SELECT città, COUNT(codAppartamento)
  FROM APPARTAMENTO
  WHERE numeroCamere >= 2
  AND prezzoGiornaliero BETWEEN 100 AND 150
  GROUP BY città

  sulla relazione:

  APPARTAMENTO (codAppartamento, indirizzo, città, mq, numeroCamere, referente, telefono, prezzoGiornaliero)

  sulla quale sono costruiti due indici: uno clustered su città ($NL_{città}$ = 171) e uno unclustered su numeroCamere ($NL_{numeroCamere}$ = 170).

  Si calcoli la selettività dei predicati e si determini il miglior piano di accesso per la risoluzione della query tenendo conto dei seguenti dati:

  NP = 4500, NT = 30000
  $NK_{città}$ = 25, $NK_{numeroCamere}$ = 4
  prezzoGiornaliero ∈ [50, 200], mq∈ [35,100], numeroCamere ∈[2,5]
  Len(città) = 20 byte, Len(mq) = 4 byte, Len(numeroCamere) = 4 byte, Len(codAppartamento) = 4 byte
  Len(prezzoGiornaliero) = 4 byte, Len(TID) = 4 byte
  D = 1Kb, u=0.69

  Si ipotizzi di utilizzare l'algoritmo Sort-Merge a Z=3 vie per l'eventuale ordinamento del risultato.
  Si indichi infine il numero atteso di tuple nel risultato della query.

Svolgimento

$$\boldsymbol{f(p1) = 1} \qquad \boldsymbol{f(p2) = \frac{150-100}{max(prezzoGiornaliero)-min(prezzoGiornaliero)} = \frac{150-100}{200-50} = \frac{1}{3}}$$

$$\boldsymbol{ET = NT \times f(p1) \times f(p2) = 30000 \times \frac{1}{3} = 10000}$$

$$NP_R = \left\lceil \frac{ET \times len(select\ list)}{D} \right\rceil = \left\lceil \frac{10000 \times 24}{1024} \right\rceil = 235$$

$$C_{sort} = 2 \times NP_R \times \lceil log_Z NP_R \rceil = 2350$$

$$C_{seq} = NP = 4500$$

$C_{città}$: non conviene perché non ci sono predicati su città i il file dati è già ordinato su questo attributo (per il group by).

$C_{numeroCamere}$**:** non conviene perché la selettività su numeroCamere è 1 (tutte) e bisognerebbe anche sommare il costo di ordinamento.

Il numero atteso di tuple è $\min(NK_{città}, ET) = 25$.

Compito Basi di Dati. Tempo concesso: 2 ore **18 Febbraio 2013 Matricola: _____**

Esercizio 3

Si consideri il seguente schema relazionale e le relative istanze di esempio:

PRODOTTO (Categoria, Sottocategoria, Prezzo)

Istanze di esempio:

PRODOTTO
(Olio, Lampante, 2.36)
(Olio, Vergine, 2.56)
(Olio, Extravergine, 3.09)
(Caffè, Arabica, 17.81)
(Caffè, Robusta, 10.32)
(Zucchero, Barbabietola, 1.15)
(Zucchero, Canna bianco, 2.12)
(Zucchero, Canna grezzo, 1.90)
(The, Verde, 26.22)
(The, Nero, 40.00)
(The, Oolong, 49.80)

1. Dato il precedente schema relazionale:

   a) Scrivere un'espressione di algebra relazionale che visualizzi i nomi delle sottocategorie dei prodotti di categoria 'Olio'.

2. Dato il precedente schema relazionale:

   a) Scrivere una query SQL che visualizzi, per ogni categoria, il prodotto di prezzo minore.
   (Il risultato deve contenere le voci Categoria, Sottocategoria, Prezzo).

   b) Scrivere una query SQL che visualizzi, per ogni categoria, i due prodotti di prezzo minore.
   (Il risultato deve contenere le voci Categoria, Sottocategoria, Prezzo).

   c) Scrivere un'espressione LINQ che selezioni le categorie che contengono almeno 2 prodotti, visualizzandole in ordine decrescente rispetto al numero di prodotti. Il risultato deve contenere le voci (Categoria, NumeroProdotti).

   Opzionale: Scrivere una query SQL che visualizzi, per ogni categoria, i due prodotti di prezzo minore, senza l'utilizzo di query innestate, assumendo che sulla base di dati esistano solo le 4 categorie riportate nelle istanze di esempio (Olio, Caffè, Zucchero, The). (Il risultato deve contenere le voci Categoria, Sottocategoria, Prezzo).

1. $Sottocategorie = \pi_{Sottocategoria}(\sigma_{Categoria='Olio'}(PRODOTTO))$

2.
a)

```
SELECT P1.Categoria, P1.Sottocategoria, P1.Prezzo
FROM PRODOTTO AS P1
WHERE P1.Prezzo = (
          SELECT MIN(Prezzo)
          FROM PRODOTTO AS P2
          WHERE P2.Categoria = P1.Categoria
          )
```

b)

```
SELECT P1.Categoria, P1.Sottocategoria, P1.Prezzo
FROM PRODOTTO AS P1
WHERE  ( SELECT COUNT(*)
            FROM PRODOTTO AS P2
            WHERE P2.Categoria = P1.Categoria
            AND P2.Prezzo <= P1.prezzo
          ) <= 2
```

```
c) var result =
    (
       from p in PRODOTTO
       group p by new { Categoria } into g
       where g.Count() >= 2
       order by g.Count() descending
       select new {Categoria = g.Key, NumeroProdotti = g.Count()}
    );
```

Opzionale

```
(SELECT TOP 2 *
FROM PRODOTTO
WHERE Categoria = 'Olio'
ORDER BY Prezzo)

UNION ALL

(SELECT TOP 2 *
FROM PRODOTTO
WHERE Categoria = 'Caffè'
ORDER BY Prezzo)

UNION ALL

(SELECT TOP 2 *
FROM PRODOTTO
WHERE Categoria = 'Zucchero'
ORDER BY Prezzo)

UNION ALL

(SELECT TOP 2 *
FROM PRODOTTO
WHERE Categoria = 'The'
ORDER BY Prezzo)
```