

A. Si deve realizzare una base di dati per descrivere le procedure presenti in una azienda, ai fini della costruzione di un sistema di gestione dei flussi di lavoro (workflow management system). Ogni procedura si compone di un certo numero di passi, con un passo iniziale e uno o più passi terminali. Ogni passo è caratterizzato da uno o più passi precedenti e uno o più passi successivi. I passi iniziali delle procedure non sono associati a nessun passo preliminare, così come i passi terminali non sono associati ad alcun passo successivo. Il legame tra un passo e i successivi può essere di tipo AND (richiedendo che al termine del passo vengano attivati tutti i passi successivi) o di tipo OR (richiedendo che venga attivato solo uno dei passi).

Ogni passo della procedura è caratterizzato da un insieme di possibili esecutori e da un insieme di possibili responsabili. L'esecutore può essere una persona o uno strumento software, il responsabile è sempre una persona. Per i passi svolti da uno strumento software, esiste sempre una persona responsabile.

Ogni istanza di esecuzione della procedura viene associata a una pratica, caratterizzata da un codice, una data di inizio e una data di chiusura. La data di chiusura può essere effettiva o prevista, a seconda che la procedura abbia o meno raggiunto un passo terminale. Ogni pratica in ogni passo ha un solo esecutore e un solo responsabile. Su passi diversi, la stessa pratica verrà normalmente gestita da diversi esecutori e diversi responsabili.

1. Svolgere il progetto concettuale; si ricorda di specificare un identificatore per ogni entità e cardinalità minima e massima di ogni relazione. (6 punti)
2. Svolgere il progetto logico, descrivendo le chiavi di ogni tabella e i "cammini di join". (3 punti)
3. Costruire in SQL una vista che abbia una tupla per ogni pratica che si trova in un passo eseguito da una persona. La vista deve associare al codice della pratica la matricola dell'esecutore che ha in carico la pratica. (2 punti)

B. Si ha il seguente schema relazionale:

```
SERVIZIO(NroTreno,Data,Capienza)
TRATTAPERCORSO(NroTreno,StazionePart,OraPart,StazioneArr,OraArr)
PRENOTAZIONE(NroTreno,Data,StazionePart,CodPosto,StazioneArr)
```

1. Costruire in SQL una vista che rappresenta il treno con le stazioni dove il convoglio si forma e si scioglie (sono le stazioni che compiono rispettivamente come StazionePart in una tupla di TRATTAPERCORSO, ma non compaiono come StazioneArr per lo stesso treno; analogamente per la stazione di arrivo). (3 punti)
2. Estrarre in SQL il treno che ha il massimo numero di posti prenotati alla sua stazione di arrivo (si faccia uso della vista creata al punto precedente). (2 punti)
3. Definire una regola Datalog che rappresenta tutte le coppie di stazioni servite da un treno. (2 punti)

C. Dato il seguente schema a oggetti:

```
create class Persona
  attributes Nome:string,
             Nazione:string,
             DataNasc:date,
             Sesso:char;
create class Opera
  attributes Titolo:string,
             Data:date,
             Regia:*Regista;
create class Film inherits Opera
  attributes Ambientazione:setof(Luogo:string),
             Incasso:integer;
create class Regista inherits Persona
  attributes Debutto:date;
create class Attore inherits Persona
  attributes Parti:setof(Ruoli:*Parte);
create class Parte
  attributes Personaggio:string,
             Sesso:char,
             Per:*Opera;
create class RecitaTeatrale inherits Opera
  attributes Teatri:setof(NomeTeatro:string);
```

1. Estrarre in OQL gli attori che non hanno mai diretto nulla e che hanno lavorato sia in teatro che nel cinema. (2 punti)
2. Estrarre i registi cinematografici che hanno recitato in teatro. (2 punti)

D. Si realizzi un albero B+ con  $F = 3$  ( $F$  pari al numero di valori di chiave presenti in ogni nodo) per rappresentare i seguenti valori: *limone, cipresso, ciliegio, tasso, melo, agrifoglio, acero, fico, pittosporo, cotoneaster, noce, oleandro, agave, euforbia, dracena*. Si costruisca anche un albero B per gli stessi valori e si mostri come potrebbe avvenire un inserimento che richieda uno split di un nodo. (4 punti)

E. Descrivere i principi fondamentali delle architetture client-server per applicazioni che fanno uso di DBMS. (4 punti)