

Basi di dati (nuovo ordinamento)
Sistemi informativi, primo modulo (vecchio ordinamento)
Prova scritta — 28 novembre 2002
Compito A

Tempo a disposizione: un'ora e trenta minuti. Libri chiusi.

Domanda 1 (10%)

Considerare le relazioni $R_1(A, B, C)$ e $R_2(D, E, F)$ aventi rispettivamente cardinalità N_1 e N_2 . Assumere che sia definito un vincolo di integrità referenziale fra l'attributo C di R_1 e la chiave D di R_2 . Indicare la cardinalità di ciascuno dei seguenti join (specificare l'intervallo nel quale essa può variare)

1. $R_1 \bowtie_{A=D} R_2$
2. $R_1 \bowtie_{C=D} R_2$
3. $R_1 \bowtie_{A=F} R_2$
4. $R_1 \bowtie_{B=E} R_2$

Domanda 2 (30%)

Si consideri una base di dati con le seguenti relazioni :

- *IMPIEGATI*(CodiceFiscale, Cognome, Nome, DataNascita, Dipartimento, Stipendio) con vincolo di integrità referenziale fra *Dipartimento* e la relazione *DIPARTIMENTI*;
- *DIPARTIMENTI*(Codice, Nome, Sede)
- *PROGETTI*(Sigla, Titolo, Valore)
- *PARTECIPAZIONE*(Impiegato, Progetto, Data) con vincoli di integrità referenziale fra *Progetto* e la relazione *PROGETTI* e fra *Impiegato* e la relazione *IMPIEGATI*

Con riferimento a tale base di dati:

1. definire in SQL la vista V come join delle relazioni *IMPIEGATI*, *PARTECIPAZIONE*, *PROGETTI*;
2. formulare in algebra relazionale l'interrogazione che trova gli impiegati che partecipano ad almeno due progetti, mostrando, per ciascuno, il codice fiscale dell'impiegato e i titoli dei due progetti;
3. formulare in SQL la stessa interrogazione di cui al punto precedente;
4. formulare in SQL l'interrogazione che mostra, per ogni dipartimento, il nome (del dipartimento), il numero di impiegati che appartengono ad esso e il loro stipendio medio.

Domanda 3 (30%)

Definire uno schema Entity-Relationship che descriva informazioni relative a sale cinematografiche di una città, secondo le seguenti specifiche:

- Ogni cinema ha un nome che lo identifica univocamente, un indirizzo e un numero di telefono. Un cinema è organizzato in più sale, ognuna delle quali ha un codice che la distingue (nell'ambito del cinema) e un numero fissato di posti.
- Per ogni sala interessa la programmazione di una sola giornata (quella odierna, senza traccia di quelle passate e future) che consiste in un elenco di proiezioni di film (eventualmente anche diversi), ognuna delle quali ha un orario di inizio.
- Per ogni film si registrano il titolo, il genere (codice e nome descrittivo), la nazionalità (una semplice stringa) e il regista (con codice identificativo, nome, cognome e anno di nascita).

Domanda 4 (15%)

Estendere lo schema concettuale proposto in risposta alla domanda precedente per rappresentare anche le seguenti specifiche:

- i posti di ciascuna sala sono numerati;
- per ogni proiezione è possibile effettuare prenotazioni, ognuna delle quali ha un codice identificativo, un nominativo e un insieme di posti.

Domanda 5 (15%)

Con riferimento allo schema mostrato nella domanda 2, scrivere un metodo Java con JDBC (o un frammento di programma in SQL immerso in un linguaggio o pseudolinguaggio di programmazione) che inserisca un impiegato con tutti i dati (letti da input o passati come parametri), verificando l'esistenza del dipartimento, con rifiuto dell'operazione in caso negativo. Assumere, per semplicità, che il sistema non supporti i vincoli di riferimento.

Basi di dati (nuovo ordinamento)
Sistemi informativi, primo modulo (vecchio ordinamento)
Prova scritta — 28 novembre 2002
Compito B

Tempo a disposizione: un'ora e trenta minuti. Libri chiusi.

Domanda 1 (10%)

Considerare le relazioni $R_1(A, B, C)$ e $R_2(D, E, F)$ aventi rispettivamente cardinalità K_1 e K_2 . Assumere che sia definito un vincolo di integrità referenziale fra l'attributo C di R_1 e la chiave D di R_2 . Indicare la cardinalità di ciascuno dei seguenti join (specificare l'intervallo nel quale essa può variare)

1. $R_1 \bowtie_{C=D} R_2$
2. $R_1 \bowtie_{A=F} R_2$
3. $R_1 \bowtie_{B=E} R_2$
4. $R_1 \bowtie_{A=D} R_2$

Domanda 2 (30%)

Si consideri una base di dati con le seguenti relazioni :

- *IMPIEGATI*(CodiceFiscale, Cognome, Nome, DataNascita, Progetto, Stipendio) con vincolo di integrità referenziale fra *Progetto* e la relazione *PROGETTI*;
- *PROGETTI*(Codice, Nome, Sede)
- *DIPARTIMENTI*(Sigla, Titolo, Budget)
- *AFFERENZA*(Impiegato, Dipartimento, Data) con vincoli di integrità referenziale fra *Dipartimento* e la relazione *DIPARTIMENTI* e fra *Impiegato* e la relazione *IMPIEGATI*

Con riferimento a tale base di dati:

1. definire in SQL la vista V come join delle relazioni *IMPIEGATI*, *AFFERENZA*, *DIPARTIMENTI*;
2. formulare in algebra relazionale l'interrogazione che trova gli impiegati che afferiscono ad almeno due dipartimenti, mostrando, per ciascuno, il codice fiscale dell'impiegato e i titoli dei due dipartimenti;
3. formulare in SQL la stessa interrogazione di cui al punto precedente;
4. formulare in SQL l'interrogazione che mostra, per ogni progetto, il nome (del progetto), il numero di impiegati che partecipano ad esso e il loro stipendio medio.

Domanda 3 (30%)

Definire uno schema Entity-Relationship che descriva informazioni relative a sale cinematografiche di un insieme di città, secondo le seguenti specifiche:

- Ogni cinema ha un nome che lo identifica nell'ambito della città, un indirizzo e un numero di telefono. Un cinema è organizzato in più sale, ognuna delle quali ha un codice che la distingue (nell'ambito del cinema) e un numero fissato di posti.
- Per ogni sala interessa la programmazione di una sola giornata (quella odierna, senza traccia di quelle passate e future) che consiste in un elenco di proiezioni di uno stesso film, ognuna delle quali ha un orario di inizio.
- Per ogni film si registrano il titolo, il genere (una semplice stringa), la nazionalità (codice e nome descrittivo) e gli attori principali (con codice identificativo, nome, cognome e anno di nascita).

Domanda 4 (15%)

Estendere lo schema concettuale proposto in risposta alla domanda precedente per rappresentare anche le seguenti specifiche:

- i posti di ciascuna sala sono numerati;
- per ogni proiezione è possibile effettuare prenotazioni, ognuna delle quali ha un codice identificativo, un nominativo e un insieme di posti.

Domanda 5 (15%)

Con riferimento allo schema mostrato nella domanda 2, scrivere un metodo Java con JDBC (o un frammento di programma in SQL immerso in un linguaggio o pseudolinguaggio di programmazione) che inserisca un impiegato con tutti i dati (letti da input o passati come parametri), verificando l'esistenza del progetto, con rifiuto dell'operazione in caso negativo. Assumere, per semplicità, che il sistema non supporti i vincoli di riferimento.

Basi di dati (nuovo ordinamento)
Sistemi informativi, primo modulo (vecchio ordinamento)
Prova scritta — 28 novembre 2002
Compito C

Tempo a disposizione: un'ora e trenta minuti. Libri chiusi.

Domanda 1 (10%)

Considerare le relazioni $R_1(A, B, C)$ e $R_2(D, E, F)$ aventi rispettivamente cardinalità C_1 e C_2 . Assumere che sia definito un vincolo di integrità referenziale fra l'attributo C di R_1 e la chiave D di R_2 . Indicare la cardinalità di ciascuno dei seguenti join (specificare l'intervallo nel quale essa può variare)

1. $R_1 \bowtie_{A=F} R_2$
2. $R_1 \bowtie_{A=D} R_2$
3. $R_1 \bowtie_{C=D} R_2$
4. $R_1 \bowtie_{B=E} R_2$

Domanda 2 (30%)

Si consideri una base di dati con le seguenti relazioni :

- *IMPIEGATI*(CodiceFiscale, *Cognome*, *Nome*, *DataNascita*, *Divisione*, *Stipendio*) con vincolo di integrità referenziale fra *Divisione* e la relazione *DIVISIONI*;
- *DIVISIONI*(Codice, *Nome*, *Sede*)
- *PROGETTI*(Sigla, *Titolo*, *Valore*)
- *PARTECIPAZIONE*(Impiegato, Progetto, *Data*) con vincoli di integrità referenziale fra *Progetto* e la relazione *PROGETTI* e fra *Impiegato* e la relazione *IMPIEGATI*

Con riferimento a tale base di dati:

1. definire in SQL la vista V come join delle relazioni *IMPIEGATI*, *PARTECIPAZIONE*, *PROGETTI*;
2. formulare in algebra relazionale l'interrogazione che trova gli impiegati che partecipano ad almeno due progetti, mostrando, per ciascuno, il codice fiscale dell'impiegato e i titoli dei due progetti;
3. formulare in SQL la stessa interrogazione di cui al punto precedente;
4. formulare in SQL l'interrogazione che mostra, per ogni divisione, il nome (della divisione), il numero di impiegati che appartengono ad essa e il loro stipendio medio.

Domanda 3 (30%)

Definire uno schema Entity-Relationship che descriva informazioni relative a sale cinematografiche di un insieme di città, secondo le seguenti specifiche:

- Ogni cinema ha un nome che lo identifica nell'ambito della città, un indirizzo e un numero di telefono. Un cinema è organizzato in più sale, ognuna delle quali ha un codice che la distingue (nell'ambito del cinema) e un numero fissato di posti.
- Per ogni sala interessa la programmazione di una sola giornata (quella odierna, senza traccia di quelle passate e future) che consiste in un elenco di proiezioni di film (eventualmente anche diversi), ognuna delle quali ha un orario di inizio.
- Per ogni film si registrano il titolo, il genere (codice e nome descrittivo), la nazionalità (una semplice stringa) e il regista (con codice identificativo, nome, cognome e anno di nascita).

Domanda 4 (15%)

Estendere lo schema concettuale proposto in risposta alla domanda precedente per rappresentare anche le seguenti specifiche:

- i posti di ciascuna sala sono numerati;
- per ogni proiezione è possibile effettuare prenotazioni, ognuna delle quali ha un codice identificativo, un nominativo e un insieme di posti.

Domanda 5 (15%)

Con riferimento allo schema mostrato nella domanda 2, scrivere un metodo Java con JDBC (o un frammento di programma in SQL immerso in un linguaggio o pseudolinguaggio di programmazione) che inserisca un impiegato con tutti i dati (letti da input o passati come parametri), verificando l'esistenza della divisione, con rifiuto dell'operazione in caso negativo. Assumere, per semplicità, che il sistema non supporti i vincoli di riferimento.

Basi di dati (nuovo ordinamento)
Sistemi informativi, primo modulo (vecchio ordinamento)
Prova scritta — 28 novembre 2002
Compito D

Tempo a disposizione: un'ora e trenta minuti. Libri chiusi.

Domanda 1 (10%)

Considerare le relazioni $R_1(\underline{A}, B, C)$ e $R_2(\underline{D}, E, F)$ aventi rispettivamente cardinalità M_1 e M_2 . Assumere che sia definito un vincolo di integrità referenziale fra l'attributo C di R_1 e la chiave D di R_2 . Indicare la cardinalità di ciascuno dei seguenti join (specificare l'intervallo nel quale essa può variare)

1. $R_1 \bowtie_{B=E} R_2$
2. $R_1 \bowtie_{C=D} R_2$
3. $R_1 \bowtie_{A=D} R_2$
4. $R_1 \bowtie_{A=F} R_2$

Domanda 2 (30%)

Si consideri una base di dati con le seguenti relazioni :

- $IMPIEGATI(\underline{CodiceFiscale}, \underline{Cognome}, \underline{Nome}, \underline{DataNascita}, \underline{Progetto}, \underline{Stipendio})$ con vincolo di integrità referenziale fra $Progetto$ e la relazione $PROGETTI$;
- $PROGETTI(\underline{Codice}, \underline{Nome}, \underline{Sede})$
- $DIVISIONI(\underline{Sigla}, \underline{Titolo}, \underline{Budget})$
- $AFFERENZA(\underline{Impiegato}, \underline{Divisione}, \underline{Data})$ con vincoli di integrità referenziale fra $Divisione$ e la relazione $DIVISIONI$ e fra $Impiegato$ e la relazione $IMPIEGATI$

Con riferimento a tale base di dati:

1. definire in SQL la vista V come join delle relazioni $IMPIEGATI$, $AFFERENZA$, $DIVISIONI$;
2. formulare in algebra relazionale l'interrogazione che trova gli impiegati che afferiscono ad almeno due divisioni, mostrando, per ciascuno, il codice fiscale dell'impiegato e i titoli delle due divisioni;
3. formulare in SQL la stessa interrogazione di cui al punto precedente;
4. formulare in SQL l'interrogazione che mostra, per ogni progetto, il nome (del progetto), il numero di impiegati che partecipano ad esso e il loro stipendio medio.

Domanda 3 (30%)

Definire uno schema Entity-Relationship che descriva informazioni relative a sale cinematografiche di una città, secondo le seguenti specifiche:

- Ogni cinema ha un nome che lo identifica univocamente, un indirizzo e un numero di telefono. Un cinema è organizzato in più sale, ognuna delle quali ha un codice che la distingue (nell'ambito del cinema) e un numero fissato di posti.
- Per ogni sala interessa la programmazione di una sola giornata (quella odierna, senza traccia di quelle passate e future) che consiste in un elenco di proiezioni di film (eventualmente anche diversi), ognuna delle quali ha un orario di inizio.
- Per ogni film si registrano il titolo, il genere (una semplice stringa), la nazionalità (codice e nome descrittivo) e gli attori principali (con codice identificativo, nome, cognome e anno di nascita).

Domanda 4 (15%)

Estendere lo schema concettuale proposto in risposta alla domanda precedente per rappresentare anche le seguenti specifiche:

- i posti di ciascuna sala sono numerati;
- per ogni proiezione è possibile effettuare prenotazioni, ognuna delle quali ha un codice identificativo, un nominativo e un insieme di posti.

Domanda 5 (15%)

Con riferimento allo schema mostrato nella domanda 2, scrivere un metodo Java con JDBC (o un frammento di programma in SQL immerso in un linguaggio o pseudolinguaggio di programmazione) che inserisca un impiegato con tutti i dati (letti da input o passati come parametri), verificando l'esistenza del progetto, con rifiuto dell'operazione in caso negativo. Assumere, per semplicità, che il sistema non supporti i vincoli di riferimento.

Basi di dati (nuovo ordinamento)
Sistemi informativi, primo modulo (vecchio ordinamento)
Prova scritta — 28 novembre 2002

Compito E

Tempo a disposizione: un'ora e trenta minuti. Libri chiusi.

Domanda 1 (10%)

Considerare le relazioni $R_1(\underline{A}, B, C)$ e $R_2(\underline{D}, E, F)$ aventi rispettivamente cardinalità M_1 e M_2 . Assumere che sia definito un vincolo di integrità referenziale fra l'attributo C di R_1 e la chiave D di R_2 . Indicare la cardinalità di ciascuno dei seguenti join (specificare l'intervallo nel quale essa può variare)

1. $R_1 \bowtie_{B=E} R_2$
2. $R_1 \bowtie_{C=D} R_2$
3. $R_1 \bowtie_{A=D} R_2$
4. $R_1 \bowtie_{A=F} R_2$

Domanda 2 (30%)

Si consideri una base di dati con le seguenti relazioni :

- $IMPIEGATI(\underline{CodiceFiscale}, \underline{Cognome}, \underline{Nome}, \underline{DataNascita}, \underline{Progetto}, \underline{Stipendio})$ con vincolo di integrità referenziale fra $Progetto$ e la relazione $PROGETTI$;
- $PROGETTI(\underline{Codice}, \underline{Nome}, \underline{Sede})$
- $DIVISIONI(\underline{Sigla}, \underline{Titolo}, \underline{Budget})$
- $AFFERENZA(\underline{Impiegato}, \underline{Divisione}, \underline{Data})$ con vincoli di integrità referenziale fra $Divisione$ e la relazione $DIVISIONI$ e fra $Impiegato$ e la relazione $IMPIEGATI$

Con riferimento a tale base di dati:

1. definire in SQL la vista V come join delle relazioni $IMPIEGATI$, $AFFERENZA$, $DIVISIONI$;
2. formulare in algebra relazionale l'interrogazione che trova gli impiegati che afferiscono ad almeno due divisioni, mostrando, per ciascuno, il codice fiscale dell'impiegato e i titoli delle due divisioni;
3. formulare in SQL la stessa interrogazione di cui al punto precedente;
4. formulare in SQL l'interrogazione che mostra, per ogni progetto, il nome (del progetto), il numero di impiegati che partecipano ad esso e il loro stipendio medio.

Domanda 3 (30%)

Definire uno schema Entity-Relationship che descriva informazioni relative a sale cinematografiche di una città, secondo le seguenti specifiche:

- Ogni cinema ha un nome che lo identifica univocamente, un indirizzo e un numero di telefono. Un cinema è organizzato in più sale, ognuna delle quali ha un codice che la distingue (nell'ambito del cinema) e un numero fissato di posti.
- Per ogni sala interessa la programmazione di una sola giornata (quella odierna, senza traccia di quelle passate e future) che consiste in un elenco di proiezioni di film (eventualmente anche diversi), ognuna delle quali ha un orario di inizio.
- Per ogni film si registrano il titolo, il genere (codice e nome descrittivo), la nazionalità (una semplice stringa) e il regista (con codice identificativo, nome, cognome e anno di nascita).

Domanda 4 (15%)

Estendere lo schema concettuale proposto in risposta alla domanda precedente per rappresentare anche le seguenti specifiche:

- i posti di ciascuna sala sono numerati;
- per ogni proiezione è possibile effettuare prenotazioni, ognuna delle quali ha un codice identificativo, un nominativo e un insieme di posti.

Domanda 5 (15%)

Con riferimento allo schema mostrato nella domanda 2, scrivere un metodo Java con JDBC (o un frammento di programma in SQL immerso in un linguaggio o pseudolinguaggio di programmazione) che inserisca un impiegato con tutti i dati (letti da input o passati come parametri), verificando l'esistenza del progetto, con rifiuto dell'operazione in caso negativo. Assumere, per semplicità, che il sistema non supporti i vincoli di riferimento.

Basi di dati (nuovo ordinamento)
Sistemi informativi, primo modulo (vecchio ordinamento)
Prova scritta — 28 novembre 2002
Compito F

Tempo a disposizione: un'ora e trenta minuti. Libri chiusi.

Domanda 1 (10%)

Considerare le relazioni $R_1(A, B, C)$ e $R_2(D, E, F)$ aventi rispettivamente cardinalità N_1 e N_2 . Assumere che sia definito un vincolo di integrità referenziale fra l'attributo C di R_1 e la chiave D di R_2 . Indicare la cardinalità di ciascuno dei seguenti join (specificare l'intervallo nel quale essa può variare)

1. $R_1 \bowtie_{A=D} R_2$
2. $R_1 \bowtie_{C=D} R_2$
3. $R_1 \bowtie_{A=F} R_2$
4. $R_1 \bowtie_{B=E} R_2$

Domanda 2 (30%)

Si consideri una base di dati con le seguenti relazioni :

- *IMPIEGATI*(*CodiceFiscale*, *Cognome*, *Nome*, *DataNascita*, *Dipartimento*, *Stipendio*) con vincolo di integrità referenziale fra *Dipartimento* e la relazione *DIPARTIMENTI*;
- *DIPARTIMENTI*(*Codice*, *Nome*, *Sede*)
- *PROGETTI*(*Sigla*, *Titolo*, *Valore*)
- *PARTECIPAZIONE*(*Impiegato*, *Progetto*, *Data*) con vincoli di integrità referenziale fra *Progetto* e la relazione *PROGETTI* e fra *Impiegato* e la relazione *IMPIEGATI*

Con riferimento a tale base di dati:

1. definire in SQL la vista V come join delle relazioni *IMPIEGATI*, *PARTECIPAZIONE*, *PROGETTI*;
2. formulare in algebra relazionale l'interrogazione che trova gli impiegati che partecipano ad almeno due progetti, mostrando, per ciascuno, il codice fiscale dell'impiegato e i titoli dei due progetti;
3. formulare in SQL la stessa interrogazione di cui al punto precedente;
4. formulare in SQL l'interrogazione che mostra, per ogni dipartimento, il nome (del dipartimento), il numero di impiegati che appartengono ad esso e il loro stipendio medio.

Domanda 3 (30%)

Definire uno schema Entity-Relationship che descriva informazioni relative a sale cinematografiche di un insieme di città, secondo le seguenti specifiche:

- Ogni cinema ha un nome che lo identifica nell'ambito della città, un indirizzo e un numero di telefono. Un cinema è organizzato in più sale, ognuna delle quali ha un codice che la distingue (nell'ambito del cinema) e un numero fissato di posti.
- Per ogni sala interessa la programmazione di una sola giornata (quella odierna, senza traccia di quelle passate e future) che consiste in un elenco di proiezioni di uno stesso film, ognuna delle quali ha un orario di inizio.
- Per ogni film si registrano il titolo, il genere (una semplice stringa), la nazionalità (codice e nome descrittivo) e gli attori principali (con codice identificativo, nome, cognome e anno di nascita).

Domanda 4 (15%)

Estendere lo schema concettuale proposto in risposta alla domanda precedente per rappresentare anche le seguenti specifiche:

- i posti di ciascuna sala sono numerati;
- per ogni proiezione è possibile effettuare prenotazioni, ognuna delle quali ha un codice identificativo, un nominativo e un insieme di posti.

Domanda 5 (15%)

Con riferimento allo schema mostrato nella domanda 2, scrivere un metodo Java con JDBC (o un frammento di programma in SQL immerso in un linguaggio o pseudolinguaggio di programmazione) che inserisca un impiegato con tutti i dati (letti da input o passati come parametri), verificando l'esistenza del dipartimento, con rifiuto dell'operazione in caso negativo. Assumere, per semplicità, che il sistema non supporti i vincoli di riferimento.

Basi di dati (nuovo ordinamento)
Sistemi informativi, primo modulo (vecchio ordinamento)
Prova scritta — 28 novembre 2002
Compito G

Tempo a disposizione: un'ora e trenta minuti. Libri chiusi.

Domanda 1 (10%)

Considerare le relazioni $R_1(A, B, C)$ e $R_2(D, E, F)$ aventi rispettivamente cardinalità K_1 e K_2 . Assumere che sia definito un vincolo di integrità referenziale fra l'attributo C di R_1 e la chiave D di R_2 . Indicare la cardinalità di ciascuno dei seguenti join (specificare l'intervallo nel quale essa può variare)

1. $R_1 \bowtie_{C=D} R_2$
2. $R_1 \bowtie_{A=F} R_2$
3. $R_1 \bowtie_{B=E} R_2$
4. $R_1 \bowtie_{A=D} R_2$

Domanda 2 (30%)

Si consideri una base di dati con le seguenti relazioni :

- *IMPIEGATI*(CodiceFiscale, Cognome, Nome, DataNascita, Progetto, Stipendio) con vincolo di integrità referenziale fra *Progetto* e la relazione *PROGETTI*;
- *PROGETTI*(Codice, Nome, Sede)
- *DIPARTIMENTI*(Sigla, Titolo, Budget)
- *AFFERENZA*(Impiegato, Dipartimento, Data) con vincoli di integrità referenziale fra *Dipartimento* e la relazione *DIPARTIMENTI* e fra *Impiegato* e la relazione *IMPIEGATI*

Con riferimento a tale base di dati:

1. definire in SQL la vista V come join delle relazioni *IMPIEGATI*, *AFFERENZA*, *DIPARTIMENTI*;
2. formulare in algebra relazionale l'interrogazione che trova gli impiegati che afferiscono ad almeno due dipartimenti, mostrando, per ciascuno, il codice fiscale dell'impiegato e i titoli dei due dipartimenti;
3. formulare in SQL la stessa interrogazione di cui al punto precedente;
4. formulare in SQL l'interrogazione che mostra, per ogni progetto, il nome (del progetto), il numero di impiegati che partecipano ad esso e il loro stipendio medio.

Domanda 3 (30%)

Definire uno schema Entity-Relationship che descriva informazioni relative a sale cinematografiche di un insieme di città, secondo le seguenti specifiche:

- Ogni cinema ha un nome che lo identifica nell'ambito della città, un indirizzo e un numero di telefono. Un cinema è organizzato in più sale, ognuna delle quali ha un codice che la distingue (nell'ambito del cinema) e un numero fissato di posti.
- Per ogni sala interessa la programmazione di una sola giornata (quella odierna, senza traccia di quelle passate e future) che consiste in un elenco di proiezioni di film (eventualmente anche diversi), ognuna delle quali ha un orario di inizio.
- Per ogni film si registrano il titolo, il genere (codice e nome descrittivo), la nazionalità (una semplice stringa) e il regista (con codice identificativo, nome, cognome e anno di nascita).

Domanda 4 (15%)

Estendere lo schema concettuale proposto in risposta alla domanda precedente per rappresentare anche le seguenti specifiche:

- i posti di ciascuna sala sono numerati;
- per ogni proiezione è possibile effettuare prenotazioni, ognuna delle quali ha un codice identificativo, un nominativo e un insieme di posti.

Domanda 5 (15%)

Con riferimento allo schema mostrato nella domanda 2, scrivere un metodo Java con JDBC (o un frammento di programma in SQL immerso in un linguaggio o pseudolinguaggio di programmazione) che inserisca un impiegato con tutti i dati (letti da input o passati come parametri), verificando l'esistenza del progetto, con rifiuto dell'operazione in caso negativo. Assumere, per semplicità, che il sistema non supporti i vincoli di riferimento.

Basi di dati (nuovo ordinamento)
Sistemi informativi, primo modulo (vecchio ordinamento)
Prova scritta — 28 novembre 2002
Compito H

Tempo a disposizione: un'ora e trenta minuti. Libri chiusi.

Domanda 1 (10%)

Considerare le relazioni $R_1(A, B, C)$ e $R_2(D, E, F)$ aventi rispettivamente cardinalità C_1 e C_2 . Assumere che sia definito un vincolo di integrità referenziale fra l'attributo C di R_1 e la chiave D di R_2 . Indicare la cardinalità di ciascuno dei seguenti join (specificare l'intervallo nel quale essa può variare)

1. $R_1 \bowtie_{A=F} R_2$
2. $R_1 \bowtie_{A=D} R_2$
3. $R_1 \bowtie_{C=D} R_2$
4. $R_1 \bowtie_{B=E} R_2$

Domanda 2 (30%)

Si consideri una base di dati con le seguenti relazioni :

- *IMPIEGATI*(CodiceFiscale, *Cognome*, *Nome*, *DataNascita*, *Divisione*, *Stipendio*) con vincolo di integrità referenziale fra *Divisione* e la relazione *DIVISIONI*;
- *DIVISIONI*(Codice, *Nome*, *Sede*)
- *PROGETTI*(Sigla, *Titolo*, *Valore*)
- *PARTECIPAZIONE*(Impiegato, Progetto, *Data*) con vincoli di integrità referenziale fra *Progetto* e la relazione *PROGETTI* e fra *Impiegato* e la relazione *IMPIEGATI*

Con riferimento a tale base di dati:

1. definire in SQL la vista V come join delle relazioni *IMPIEGATI*, *PARTECIPAZIONE*, *PROGETTI*;
2. formulare in algebra relazionale l'interrogazione che trova gli impiegati che partecipano ad almeno due progetti, mostrando, per ciascuno, il codice fiscale dell'impiegato e i titoli dei due progetti;
3. formulare in SQL la stessa interrogazione di cui al punto precedente;
4. formulare in SQL l'interrogazione che mostra, per ogni divisione, il nome (della divisione), il numero di impiegati che appartengono ad essa e il loro stipendio medio.

Domanda 3 (30%)

Definire uno schema Entity-Relationship che descriva informazioni relative a sale cinematografiche di una città, secondo le seguenti specifiche:

- Ogni cinema ha un nome che lo identifica univocamente, un indirizzo e un numero di telefono. Un cinema è organizzato in più sale, ognuna delle quali ha un codice che la distingue (nell'ambito del cinema) e un numero fissato di posti.
- Per ogni sala interessa la programmazione di una sola giornata (quella odierna, senza traccia di quelle passate e future) che consiste in un elenco di proiezioni di film (eventualmente anche diversi), ognuna delle quali ha un orario di inizio.
- Per ogni film si registrano il titolo, il genere (una semplice stringa), la nazionalità (codice e nome descrittivo) e gli attori principali (con codice identificativo, nome, cognome e anno di nascita).

Domanda 4 (15%)

Estendere lo schema concettuale proposto in risposta alla domanda precedente per rappresentare anche le seguenti specifiche:

- i posti di ciascuna sala sono numerati;
- per ogni proiezione è possibile effettuare prenotazioni, ognuna delle quali ha un codice identificativo, un nominativo e un insieme di posti.

Domanda 5 (15%)

Con riferimento allo schema mostrato nella domanda 2, scrivere un metodo Java con JDBC (o un frammento di programma in SQL immerso in un linguaggio o pseudolinguaggio di programmazione) che inserisca un impiegato con tutti i dati (letti da input o passati come parametri), verificando l'esistenza della divisione, con rifiuto dell'operazione in caso negativo. Assumere, per semplicità, che il sistema non supporti i vincoli di riferimento.