

# Esame di Basi di Dati

## A.A. 2022/2023 – Appello del 04/07/2023

### Problema 1

Si richiede di progettare lo schema concettuale Entità-Relazione di un'applicazione relativa a viaggi organizzati. Di ogni viaggio interessa il codice (identificativo), le persone che vi hanno partecipato e le tappe in esso svolte. Di ogni persona interessa il codice fiscale (identificativo), il nome, il cognome e la città di nascita. Di ogni viaggio si svolge al massimo una tappa al giorno e di ognuna di tali tappe interessa la data in cui si è svolta, il costo e la città principale che è stata visitata nella tappa. Inoltre, di ogni "tappa speciale" interessa anche la maggiorazione del costo e le città (almeno una) visitate in aggiunta a quella principale (queste città vengono chiamate città secondarie della tappa), ciascuna con la durata della visita in quella città e con la persona che ha ricoperto il ruolo di guida nella visita di quella città. Si noti che una persona non può ricoprire il ruolo di guida in più di una visita ad una città secondaria nella stessa tappa di un viaggio (ma in tappe diverse sì). Infine, di ogni città interessa il codice (identificativo), il nome e la regione.

### Problema 2

Si richiede di effettuare la progettazione logica per l'applicazione citata nel problema 1, tenendo conto del fatto che (i) quando si accede ad una persona si vuole sempre conoscere la sua città di nascita; (ii) quando si accede ad una tappa si vuole sapere se è una tappa speciale ed in questo caso anche quanto è la maggiorazione del costo.

### Problema 3

Riferendosi allo schema logico prodotto per il problema 2, scrivere una query SQL che per ogni persona calcoli il codice fiscale ed il numero di viaggi in cui ha ricoperto il ruolo di guida in almeno una tappa.

### Problema 4

In una base di dati  $B$ , la relazione  $\text{NodiGrafo}(\underline{\text{nodo}})$  memorizza tutti i nodi di un grafo orientato  $G$  e la relazione  $\text{ArchiGrafo}(\underline{\text{partenza}}, \underline{\text{arrivo}})$  memorizza tutti gli archi di  $G$ , dove un arco da un nodo  $n_1$  ad un nodo  $n_2$  è rappresentato dalla tupla  $t = \langle n_1, n_2 \rangle$  nella relazione  $\text{ArchiGrafo}$ , dove  $t.\text{partenza} = n_1$  e  $t.\text{arrivo} = n_2$ . Sappiamo che  $B$  soddisfa sia il vincolo di foreign key (integrità referenziale) da  $\text{partenza}$  di  $\text{ArchiGrafo}$  a  $\text{nodo}$  di  $\text{NodiGrafo}$  sia il vincolo di foreign key da  $\text{arrivo}$  di  $\text{ArchiGrafo}$  a  $\text{nodo}$  di  $\text{NodiGrafo}$ . Si noti che se  $G$  comprende l'arco da  $n_1$  a  $n_2$ , allora  $n_2$  si dice *successore di  $n_1$*  in  $G$  ed  $n_1$  si dice *predecessore di  $n_2$*  in  $G$ . Inoltre, un nodo si dice *sorgente* se ha almeno un successore e nessun predecessore. Si chiede di rispondere alle seguenti domande.

1. Scrivere una query SQL che calcoli il numero medio di predecessori dei nodi  $G$ .
2. Scrivere una query in algebra relazionale che calcoli tutti i nodi di  $G$  che hanno come predecessori solo nodi sorgente.

### Problema 5

Considerando lo schema concettuale  $S$  riportato qui sotto, effettuare la progettazione logica partendo da  $S$ , mostrando sia lo schema concettuale ristrutturato sia lo schema logico proveniente dalla traduzione diretta. Cambierebbe il risultato della progettazione logica effettuata se le indicazioni di progetto sancissero il fatto che alle istanze di  $F$  si accede prevalentemente utilizzando il valore di  $A$ ?

