

(A) CONOSCENZA TERMINOLOGICA

Dare una breve descrizione dei termini introdotti:

- Funzione hash
- Spazio delle chiavi
- Spazio degli indirizzi
- Trasformazione perfetta
- Collisioni
- Metodo di indirizzamento aperto
- Legge di scansione
- Scansione a passo unitario
- Scansione quadratica
- Metodo della divisione
- Metodo della moltiplicazione

(B) CONOSCENZA E COMPETENZA

Rispondere alle seguenti domande producendo anche qualche esempio

B1) Conoscenza

1. Quali sono le principali caratteristiche dell'organizzazione hash?
2. Come è definita una *funzione hash*?
3. Quali sono le caratteristiche di una *trasformazione perfetta*?
4. Quali caratteristiche deve avere una *funzione hash* per essere efficiente?

B2) Competenza

1. Quali sono le tecniche elementari per implementare la *formula di randomizzazione*?
2. Come si deve procedere se N (ampiezza dello spazio degli indirizzi) *non è un numero primo*?
3. Quali considerazioni si possono fare riguardo alla rappresentazione di dati in forma di *tabella*?
4. Se la *funzione di randomizzazione* usa la somma dei codici ASCII della chiave, in quale caso si verifica la collisione.

(C) ESERCIZI DI COMPrensIONE

1. In un archivio con organizzazione la posizione logica di un record viene determinata da una che serve a determinare il di ciascun record.
2. La funzione riceve in ingresso la, in genere di tipo, e la converte in un che rappresenta il del record associato.
3. La funzione *hash* ha come lo spazio delle e come lo spazio degli Se la $f(K)$ è biunivoca, si dice trasformazione In realtà la trasformazione non viene usata perché crea problemi di
4. Una trasformazione tra le più elementari è quella che fa uso del e in cui il valore di N deve essere primo.
5. La funzione **integer(K)**, detta formula di, deve essere di computazione e deve generare un valore molto più di N.
6. La tecnica è molto frequente nell'organizzazione dei dati in memoria di massa, poiché da un lato evita lo di memoria, dall'altro consente un'elevata Nelle operazioni di ricerca e aggiornamento dei dati.
7. La funzione è una funzione del tipo:

$$f: \{S_K\} \rightarrow \{0, \dots, N-1\}$$
 dove S_K è lo spazio delle e $\{0, \dots, N-1\}$ lo spazio degli
8. In una trasformazione ogni dato può essere reperito mediante un unico La trasformazione perfetta è un caso ideale, che in pratica non conviene realizzare mai poiché implica un grande consumo di
9. Il metodo della è una semplice funzione che, data la chiave K, calcola l'indirizzo come resto della divisione tra e il numero dei record N. In questo metodo è molto importante la scelta del valore di N: in generale, si ottiene un sistema efficiente se N è un numero
10. Si considerino i cognomi indicati nella tabella seguente:
 - a. assumendo come chiave i primi quattro byte del campo *Cognome*, riportare nella seconda colonna la chiave e nell'ultima colonna l'indirizzo generato sommando le posizioni alfabetiche dei caratteri.

| Cognome | Chiave | Dati del record | | | Indirizzo |
|-----------|--------|-----------------|------|------|-----------|
| Anzalone | | ... | | | |
| Bianchi | | | | | |
| Caraffa | | | | | |
| Caratelli | | | | | |
| Lori | | | | | |
| Loria | | | | | |
| Pizzi | | | | | |
| Pizzini | | | | | |
| Realacci | | | | | |
| Reali | | | | | |

- b. assumendo come chiave i primi quattro byte del campo *Cognome*, riportare nella seconda colonna la chiave e nell'ultima colonna l'indirizzo generato sommando i codici ASCII dei caratteri formanti la chiave;

| Cognome | Chiave | Dati del record | | | Indirizzo |
|-----------|--------|-----------------|--|--|-----------|
| Anzalone | | | | | |
| Bianchi | | | | | |
| Caraffa | | | | | |
| Caratelli | | | | | |
| Lori | | | | | |
| Loria | | | | | |
| Pizzi | | | | | |
| Pizzini | | | | | |
| Realacci | | | | | |
| Reali | | | | | |

- c. assumendo come chiave l'intero campo *Cognome*, riportare nella seconda colonna la chiave e nell'ultima colonna l'indirizzo generato sommando le posizioni alfabetiche dei singoli caratteri.

| Cognome | Chiave | Dati del record | | | Indirizzo |
|-----------|--------|-----------------|--|--|-----------|
| Anzalone | | | | | |
| Bianchi | | | | | |
| Caraffa | | | | | |
| Caratelli | | | | | |
| Lori | | | | | |
| Loria | | | | | |
| Pizzi | | | | | |
| Pizzini | | | | | |
| Realacci | | | | | |
| Reali | | | | | |

- d. assumendo come chiave la somma dei codici ASCII dei caratteri del campo *Nome*, riportare nella seconda colonna la chiave e nell'ultima colonna l'indirizzo generato, segnalando eventuali collisioni

| | Chiave | Dati del record | | | Indirizzo |
|---------|--------|-----------------|--|--|-----------|
| Bologna | | | | | |
| Taranto | | | | | |
| Otranto | | | | | |
| Toronto | | | | | |
| Bari | | | | | |
| Toranta | | | | | |
| Biro | | | | | |
| Bagnolo | | | | | |

11. La tecnica hash è spesso usata dai file system per ottenere gli indirizzi assoluti su disco. Supponiamo di avere chiavi numeriche aventi formato xx-yy-zz da allocare su 12 unità a disco con 600 cilindri, ciascuno di 40 tracce. Si possono utilizzare tre distinte funzioni hash, ad esempio:
 $(xx + yy + z) \bmod 12$ per individuare l'unità
 $(x + yy + z) \bmod 600$ per individuare il cilindro
 $(x + y + z) \bmod 40$ per la traccia.
 Calcolare gli indirizzi corrispondenti alle seguenti chiavi:
 a. 15-20-25
 b. 06-56-78
 c. 15-20-2
 ed indicare le eventuali collisioni
12. Supponiamo di dover codificare 1000 prodotti di magazzino avente chiave numerica compresa nell'intervallo 2000 – 2999. Individuare una possibile funzione hash che dalla chiave, generi gli indirizzi da 1 a 1000.
13. Si consideri il calcolo della seguente funzione hash:

$$h(k) = \text{integer}(k) \bmod N$$

dove **integer(k)** indica una funzione che converte la chiave k in un numero intero. Per una chiave k di tipo **Stringa** si potrebbe avere, ad esempio:

$$\mathbf{integer(k)=bin(k[0])}$$

che indica la conversione del carattere $k[0]$ in binario. Stabilire quale dovrebbe essere il valore di N per evitare le collisioni.

14. Data la serie di chiavi in prima colonna, calcolare gli indirizzi generati da ciascuna delle funzioni hash indicate.

| K | (K mod 10) + 1 | (K mod 15)+1 | (K mod 20)+1 |
|----|----------------|--------------|--------------|
| 91 | | | |
| 70 | | | |
| 25 | | | |
| 78 | | | |
| 14 | | | |
| 21 | | | |
| 9 | | | |
| 77 | | | |
| 10 | | | |
| 55 | | | |

(D) ESERCIZI DI APPLICAZIONE

- Se si vuole costruire una tabella di 20 numeri di 2 cifre decimali ciascuno, calcolare:
 - il numero di possibili valori diversi da porre in ciascuna voce della tabella;
 - il range di ciascun valore;
 - il numero di locazioni della tabella, in caso di una trasformazione perfetta;
 - il numero medio di collisioni se si dispone di una tabella di 20 locazioni.
- L'organizzazione *hash* si può applicare anche in memoria centrale per gestire ricerche e inserimenti in una tabella. Implementare una organizzazione di tale tipo con le operazioni relative.
- Implementare la funzione **integer(k)** nel caso di:
 - somma dei codici ASCII dei caratteri
 - somma delle posizioni alfabetiche
 - somma delle cifre della chiave
 - primi 4 caratteri della chiave