

Dizionari

Dizionari	<p>Un dizionario è una collezione di elementi omogenei e di tipo chiave-valore ai quali si accede mediante il campo chiave. Può essere considerato un sottotipo dell'insieme.</p> <p>Le operazioni consentite sono: crea, appartiene, inserisci, cancella e talvolta recupera e aggiorna. Si ricorre ai dizionari quando sono necessarie alcune caratteristiche degli insiemi (come l'unicità degli elementi) ma senza alcune operazioni caratteristiche come unione, intersezione etc.</p>
Rappresentazioni: Vettore Ordinato Ricerca Binaria	<p>Si definisce una relazione di ordinamento totale \leq sulle chiavi e quindi si memorizzano in ordine crescente in un vettore e si utilizza un cursore per identificare l'ultima posizione occupata. Si utilizza la ricerca binaria.</p> <p>A differenza della ricerca sequenziale, dove gli elementi vengono scanditi uno per uno, nella ricerca binaria si confronta la chiave dell'elemento da ricercare all'interno dell'array con la chiave dell'elemento nel mezzo dell'array; se la chiave è uguale la ricerca termina, altrimenti, visto che le chiavi sono ordinate in ordine crescente, si valuta se continuare la ricerca nella metà di sinistra o destra dell'array.</p>
Rappresentazioni: Hash Table	<p>L'implementazione attraverso hash table permette una maggiore efficienza data dalle caratteristiche di questa tecnica.</p> <p>L'idea alla base dell'hash table è quella di poter ritrovare il valore efficientemente, ricavando la posizione del valore dalla chiave stessa. Le operazioni di ricerca e modifica impiegheranno tempo costante, indipendentemente dalla dimensione della tabella.</p> <p>Esistono diverse varianti di hashing</p> <ul style="list-style-type: none">- Hash Statico Utilizza tabelle/strutture di dimensioni prefissate. Due possibili forme sono:<ul style="list-style-type: none">- Hash Chiuso: consente di inserire un numero limitato di valori in uno spazio di dimensione fissa. In questo spazio avremo una struttura (spesso un array) composta da <i>buckets</i>. Ci sarà quindi numero fisso di <i>bucket</i>, che potrà contenere al massimo una coppia <i><chiave, valore></i>.- Hash Aperto: non pone limiti sul numero di valori e lo spazio è di dimensione potenzialmente infinita. Anche qui vengono utilizzati dei bucket per le coppie <i><chiave, valore></i>, ma a differenza dell'hash chiuso il numero sarà indefinito.In entrambi i casi si utilizza una funzione di hashing per determinare la posizione del valore nel bucket.- Hash Dinamico Utilizza tabelle/strutture di dimensioni variabili.

Funzione di Hashing (Hash Chiuso)	<p>Come detto prima, la funzione di hashing permette di ricavare la posizione del valore associato ad una specifica chiave dalla chiave stessa.</p> <p>Sia allora:</p> <ul style="list-style-type: none"> - K: insieme di tutte le possibili chiavi distinte; - m: dimensione dello spazio; - v: vettore dei bucket di dimensione m. <p>Definiamo quindi la funzione di hashing h come:</p> $h: K \rightarrow \{1 \dots m\}$ <p>Allora $h(k)$ permette di ricavare la posizione della chiave k nel vettore v.</p> <p>Per quanto riguarda la dimensione del vettore abbiamo che, se si usa $m = K$ si ha garanzia di biunivocità e quindi di accedere direttamente alla posizione contenente la chiave.</p> <p>Il problema sorge quando K è grande: questo comporta un enorme spreco di memoria. Un buon compromesso per ovviare a questo problema è scegliere un m maggiore di 1 ma molto minore di K.</p>
Esempio di Funzione di Hashing	<p>La seguente funzione di hashing utilizza la posizione della prima e seconda lettera della chiave nell'alfabeto, sottraendo le rispettive posizioni, per determinare la posizione della chiave desiderata.</p> <p>Ad esempio:</p> <ul style="list-style-type: none"> - chiave: <i>paul</i> $p = 16, a = 1$ - m = 8 <p>Quindi $h(k) = 15$, ma visto che $15 > 8$ si utilizza l'operatore modulo: $15 \bmod 8 = 7$.</p>
Collisione	<p>Si ha una collisione quando una funzione di hashing dà lo stesso risultato con due chiavi diverse.</p> <p>Alcune possibili soluzioni a questo problema sono:</p> <ul style="list-style-type: none"> - Scansione Lineare In un'operazione come l'inserimento, questa consiste nell'usare una chiave fittizia <i>libero</i> in fase di progettazione e nell'inserire il valore nella prima posizione libera (quando la posizione indicata dalla funzione di hashing indica una posizione già occupata). Una soluzione del genere può però favorire la formazione di agglomerati. - Liste di Trabocco Si utilizza una lista nella posizione che contiene più chiavi.
Metodi di Scansione	<p>Nelle liste di trabocco si usa un metodo di scansione esterno (invece di un metodo interno come la scansione lineare).</p> <p>Altri metodi di scansione interna sono:</p> <ul style="list-style-type: none"> - Scansione Lineare - Scansione Quadratica - Scansione Pseudocasuale - Hashing Doppio <p>approfondire</p>

Scelta della Funzione di Hashing	<p>Una buona funzione di hashing deve essere:</p> <ul style="list-style-type: none">- calcolabile velocemente;- distribuire le chiavi uniformemente (in modo da evitare collisioni). <p>Inoltre, bisogna avere una soluzione al problema delle collisioni che eviti anche gli agglomerati e avere un vettore appena più grande del necessario per non rimanere mai senza spazio.</p>
---	---