

Insiemi

Specifiche, rappresentazione e confronto tra realizzazioni alternative.

Algoritmi e Strutture Dati + Lab

A.A. 18/19

Informatica

Università degli Studi di Bari "Aldo Moro"

Nicola Di Mauro

Definizione

- Un insieme è una collezione (o famiglia) di elementi (componenti o membri) di tipo omogeneo. A differenza delle liste gli elementi non sono caratterizzati da una posizione né possono apparire più di una volta.
- In matematica possono essere definiti *estensionalmente*
 - $A = \{ \text{giallo, rosso, blu} \}$
- oppure *intensionalmente* attraverso le proprietà che devono avere i componenti
 - $B = \{ \text{elementi nel collegio ba19 nel 1994} \}$
 - $C = \{ \text{numeri reali compresi tra 0 e 1} \}$
- In informatica ci riferiamo al modo estensionale

Operazioni

- Il numero di elementi ($|A|$ detto cardinalità) rappresenta la dimensione dell'insieme
 - $|A| = 3$
 - $|B|$ è finita
 - $|C|$ è infinita
- La relazione fondamentale è quella di appartenenza $x \in A$, da cui deriva l'inclusione $B \subseteq A$.
- Operazioni principali sono:
 - unione $A \cup B$
 - intersezione $A \cap B$
 - differenza $A \setminus B$

Specifica sintattica

- Tipi
 - insieme , boolean , tipoelem
- Operatori
 - Creainsieme: $() \rightarrow \text{insieme}$
 - Insiemevuoto: $(\text{insieme}) \rightarrow \text{boolean}$
 - Appartiene: $(\text{tipoelem} , \text{insieme}) \rightarrow \text{boolean}$
 - Inserisci: $(\text{tipoelem} , \text{insieme}) \rightarrow \text{insieme}$
 - Cancella: $(\text{tipoelem} , \text{insieme}) \rightarrow \text{insieme}$
 - Unione: $(\text{insieme} , \text{insieme}) \rightarrow \text{insieme}$
 - Intersezione: $(\text{insieme} , \text{insieme}) \rightarrow \text{insieme}$
 - Differenza: $(\text{insieme} , \text{insieme}) \rightarrow \text{insieme}$

Specifica semantica

- Tipi

- insieme = famiglia di insiemi costituita da elementi di tipo tipoelem
- boolean = insieme valori verità

- Operatori

- creainsieme = A
 - $\text{post} : A = \{\}$
- insiemevuoto(A) = b
 - $\text{post} : b = \text{vero se } a = \{\}, b = \text{falso altrimenti}$
- appartiene (x, A) = b
 - $\text{post} : b = \text{vero se } x \in A, b = \text{falso altrimenti}$
- inserisci (x, A) = A'
 - $\text{pre} : x \notin A$ (oppure senza precondizione)
 - $\text{post} : A' = A \cup \{x\}$ (se $x \in A : A' = A$)
- cancella (x, A) = A'
 - $\text{pre} : x \in A$ (oppure senza precondizione)
 - $\text{post} : A' = A \setminus \{x\}$ (se $x \notin A : A' = A$)

Specifica semantica /2

- $\text{unione}(A, B) = C$
 - post: $C = A \cup B$
- $\text{intersezione}(A, B) = C$
 - post: $C = A \cap B$
- $\text{differenza}(A, B) = C$
 - post: $C = A \setminus B$

Realizzazioni

- Rappresentazione con vettore booleano
 - Per linguaggi che non dispongono del tipo insieme, è possibile rappresentare un insieme A , i cui elementi siano, ad esempio, interi in $[1, n]$, attraverso un vettore booleano di n bit, il cui k -esimo valore sarà “vero” se $k \in A$ e “falso” altrimenti. (Vettore caratteristico)
 - Un'altra possibile rappresentazione si avvale di una lista i cui elementi sono quelli dell'insieme
 - in tal modo si può evitare che gli elementi siano assolutamente degli interi

Realizzazioni con liste non ordinate

- Gli elementi della lista sono quelli dell'insieme. Nel caso si usino realizzazioni con strutture dinamiche, l'occupazione di memoria è proporzionale al numero degli elementi presenti nell'insieme

```
class cella{  
    tipoelem elemento;  
    posizione successivo;  
}  
class insieme{  
    cella * posizione;  
}
```

- L'inserimento avviene in testa alla lista semplice con cui è realizzato l'insieme

Realizzazioni con liste ordinate

- Se è definita una relazione \leq di ordinamento totale sugli elementi dell'insieme, esso può essere rappresentato con una lista ordinata per valori crescenti degli elementi utilizzando due puntatori che scorrono ognuno su un insieme.
- La realizzazione degli operatori non presenta particolari difficoltà

Applicazione

- **Problema:** trovare i numeri primi appartenenti all'intervallo $2..n$ con $n > 2$
- Algoritmo (setaccio di eratostene)
 - metti tutti i numeri tra 2 e n nel “setaccio”
 - scegli e rimuovi il numero in “setaccio”
 - includi questo numero in “numeri primi”
 - rimuovi dal “setaccio” tutti i multipli di questo numero
 - se il “setaccio” non è vuoto ripeti i passi 2- 5
- Sia “setaccio” che “numeri primi” sono definibili come insiemi.

mfset

- Ci sono spesso delle applicazioni che non richiedono l'uso di tutte le operazioni
 - Tenere le registrazioni degli stipendi degli impiegati di una impresa: non interessa alcuna struttura che riguarda gli impiegati e le operazioni unione, intersezione e differenza non sono necessarie
- Implementare la tabella dei simboli
 - come è noto la tabella dei simboli di un compilatore è usata per memorizzare i nomi delle costanti, dei tipi e delle variabili di un programma sorgente
 - tradizionalmente si usano strutture particolari (tavole o dizionari).
- In questi casi si opera attraverso la struttura **mfset(merge-find-set)**

mfset /2

- Un mfset è una partizione di un insieme finito in sottoinsiemi disgiunti detti componenti.
- Le operazioni consentite permettono di :
 - Stabilire a quale componente appartiene un elemento generico
 - Unire due componenti distinte in una sola componente lasciando inalterate le componenti rimanenti
- Specifica sintattica
 - Tipi: insieme , boolean , tipoelem , mfset, componente
 - Operatori
 - creamfset: (insieme) \rightarrow mfset
 - fondi: (tipoelem,tipoelem,mfset) \rightarrow mfset
 - trova: (tipoelem,mfset) \rightarrow componente

Specifica semantica

- Tipi
 - insieme = famiglia di insiemi costituita da elementi di tipo `tipoelem`
 - `mfset` = famiglia di partizioni di insiemi di elementi di tipo `tipoelem`
 - componente = sottoinsieme di insieme, che è elemento di `mfset`
- Operatori:
 - `creamfset(A) = S`
 - `post : S` è una famiglia di $n=|A|$ componenti c_1, c_2, \dots, c_n ognuno delle quali contiene uno e un solo elemento di A e tali che $\bigcup c_i = A, 1 \leq i \leq n$
 - `fondi (x , y , S)= S'`
 - `pre : x` e `y` appartengono a componenti distinte c_x e c_y di S
 - `post : S'` è costituito da tutte le componenti che non contengono x e y e da una nuova componente ottenuta dall'unione delle due componenti c_x e c_y
 - `trova (x , S)= c`
 - `pre : x` appartiene ad una componente di S
 - `post : c` è l'identificatore della componente cui x appartiene

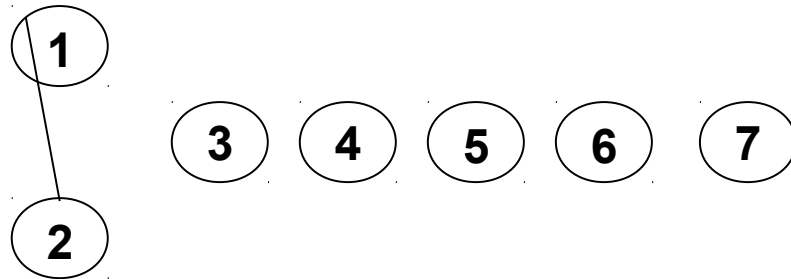
mfset /4

- In letteratura si può trovare anche un altro operatore trova teso a verificare se dati due elementi questi appartengono alla stessa componente in questo caso
 - trova: (tipoelem , tipoelem , mfset) \rightarrow boolean
 - trova (x , y , S)= b
 - pre : x e y appartengono a componenti di s
 - post : b = vero se x e y appartengono alla stessa componente, falso altrimenti
- Realizzazioni efficaci di mfset prevedono l'uso di strutture ad albero che saranno presentate più avanti. Ora è data una rappresentazione intuitiva.
- Sia $A = \{1, 2, 3, 4, 5, 6, 7\}$
 - creamfset restituisce $S = \{ [1], [2], [3], [4], [5], [6], [7] \}$
- Poiché trova(1, S) è diverso da trova(5, S), allora si può applicare la operazione di fusione
 - fondi (1, 5, S) = $\{ [2], [3], [4], [1, 5], [6], [7] \}$

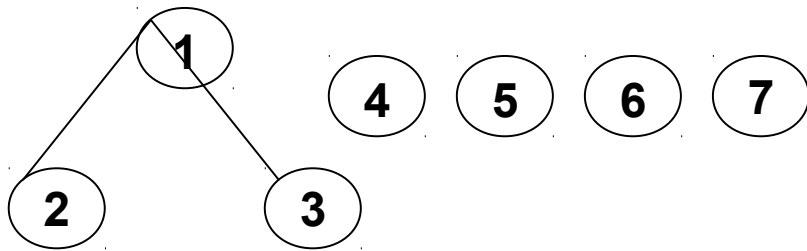
mfset /5



FONDI (1,2,S)



FONDI (1,3,S)



FONDI (5,6,S)

