

Sessione ordinaria 1992 PNI**PROBLEMA 3**

Si desidera fondere due sequenze A e B di numeri interi, non ordinate e con eventuali valori ripetuti, in un'unica sequenza C nella quale compaiono, in ordine crescente e senza ripetizioni, i valori presenti in A e in B.

Il candidato, formulate le eventuali ipotesi aggiuntive che ritiene necessarie, proponga ed illustri una procedura per risolvere il problema e la codifichi in un linguaggio di sua conoscenza.

Soluzione

La soluzione di questo problema non è, ovviamente, univoca.

Ci limitiamo a una descrizione delle procedure e degli algoritmi che concorrono alla stesura del programma.

Il problema deve essere scomposto in sottoproblemi da risolvere mediante opportuni algoritmi.

In particolare, l'algoritmo di ordinamento di una lista di numeri è un esempio classico nella programmazione insegnata nei corsi PNI, con la relativa codifica in linguaggio Pascal

Il programma principale si articola nella seguente sequenza di sottoprogrammi o procedure

1. Acquisire i dati (lunghezza di ciascuna sequenza (a e b) di numeri e i rispettivi valori
2. Fondere le due sequenze in un'unica sequenza c
3. Ordinare la sequenza c in ordine crescente
4. Eliminare gli elementi ripetuti
5. Comunicare il risultato

1. Le variabili a e b sono due array unidimensionali (vettori) di numeri interi. Le rispettive lunghezze sono variabili di tipo "integer" di cui si può fissare inizialmente il valore massimo.

La procedura di <<riempimento>> di un vettore $b[i]$ di lunghezza n è di tipo iterativo, ad esempio

```
<<Per i che va da 1 ad n Leggi  $x[i]$  >>
```

2. Una volta costruiti i due vettori, $a[i]$ di lunghezza l e $b[i]$ di lunghezza m , si costruisce un vettore unico $c[i]$ di lunghezza $l + m$ nel modo seguente

```
<<Per i che va da 1 ad  $l$  poni  $c[i] = a[i]$ 
```

```
Per i che va da 1 ad  $m$  poni  $c[l + 1] = b[i]$  >>
```

ESEMPIO – verifica procedure-

INSERISCI

$a[i]$			
5	4	1	3

$b[i]$		
2	1	5

FONDI

$c[i]$						
5	4	1	3	2	1	5

3. Per ordinare il vettore $c[i]$ di lunghezza $N = l + m$ si può utilizzare l'algoritmo di ordinamento a scambio o "bubble-sort", algoritmo molto semplice anche se poco efficiente in caso di grande numerosità dei dati. Include la procedura che scambia due elementi consecutivi,

La procedura SCAMBIA (x, y) è illustrata dal seguente algoritmo che introduce la variabile ausiliaria "appoggio"

```

inizio
  appoggio  $\leftarrow x$ 
   $x \leftarrow y$ 
   $y \leftarrow \text{appoggio}$ 
fine

```

In caso di ordinamento in senso crescente l'algoritmo bubble-sort consiste nel confrontare ogni elemento con il successivo e scambiare di posto i due valori se il primo è maggiore del secondo; ripetendo la procedura, i valori minori saliranno verso l'alto come bolle leggere.

Il vettore risulterà sicuramente ordinato in ordine crescente non appena nella scansione non sarà effettuato alcun scambio.

Algoritmo della **procedura ORDINA** (vettore $c[i]$) che introduce la variabile booleana "Scambio"

Inizio

Ripeti

 $Scambio \leftarrow falso$

Fai una scansione del vettore a partire dal primo fino al penultimo termine a e

se $c[i] > c[i + 1]$

$$\left\{ \begin{array}{l} \text{scambia } c[i] \text{ e } c[i + 1] \\ Scambio \leftarrow VERO \end{array} \right.$$

Finché

 $Scambio = falso$ **Fine**

ESEMPIO**ORDINA-** risultati di ciascuna scansione

Inizio	$c[i]$							
	5	4	1	3	2	1	5	<i>Scambio = falso</i>
$i = 1$	4	5	1	3	2	1	5	<i>Scambio = vero</i>
$i = 2$	4	1	5	3	2	1	5	<i>Scambio = vero</i>
$i = 3$	4	1	3	5	2	1	5	<i>Scambio = vero</i>
$i = 4$	4	1	3	2	5	1	5	<i>Scambio = vero</i>
$i = 5$	4	1	3	2	1	5	5	<i>Scambio = vero</i>
$i = 6$	4	1	3	2	1	5	5	
								<i>Scambio = falso</i>
$i = 1$	1	4	3	2	1	5	5	<i>Scambio = vero</i>
$i = 2$	1	3	4	2	1	5	5	<i>Scambio = vero</i>
$i = 3$	1	3	2	4	1	5	5	<i>Scambio = vero</i>
$i = 4$	1	3	2	1	4	5	5	<i>Scambio = vero</i>
$i = 5$	1	3	2	1	4	5	5	
$i = 6$	1	3	2	1	4	5	5	
								<i>Scambio = falso</i>
$i = 1$	1	3	2	1	4	5	5	
$i = 2$	1	2	3	1	4	5	5	<i>Scambio = vero</i>
$i = 3$	1	2	1	3	4	5	5	<i>Scambio = vero</i>
$i = 4$	1	2	1	3	4	5	5	
$i = 5$	1	2	1	3	4	5	5	
$i = 6$	1	2	1	3	4	5	5	
								<i>Scambio = falso</i>
$i = 1$	1	2	1	3	4	5	5	
$i = 2$	1	1	2	3	4	5	5	<i>Scambio = vero</i>
$i = 3$	1	1	2	3	4	5	5	
$i = 4$	1	1	2	3	4	5	5	
$i = 5$	1	1	2	3	4	5	5	
$i = 6$	1	1	2	3	4	5	5	
								<i>Scambio = falso</i>
$i = 1$	1	1	2	3	4	5	5	
$i = 2$	1	1	2	3	4	5	5	
$i = 3$	1	1	2	3	4	5	5	
$i = 4$	1	1	2	3	4	5	5	
$i = 5$	1	1	2	3	4	5	5	
$i = 6$	1	1	2	3	4	5	5	
Fine	1	1	2	3	4	5	5	

4. Algoritmo della Procedura UNIFICA

Per costruire, a partire dal vettore $c[i]$ ordinato, il vettore definitivo con elementi tutti distinti, accanto all'indice i che identifica gli elementi del primo vettore, introduciamo un ulteriore indice j per costruire il secondo $v[j]$.

- Si assegna al primo elemento il valore $v[1] = c[1]$ e si incrementa di un'unità l'indice $j: j = 2$
- Si confronta poi $c[1]$ con $c[2]$: se sono diversi si assegna a $v[2]$ il valore $c[2]$ e si incrementa di un'unità l'indice $j: j = 3$, altrimenti si lascia tutto invariato
- Si procede in modo analogo con gli elementi successivi, fino a penultimo valore di $c[i]$:

si confronta ogni elemento di $c[i]$ con il successivo; se sono diversi si assegna a $v[j]$ il valore $c[i + 1]$ e si incrementa di un'unità l'indice $j: j = j + 1$

Inizio

$$\begin{cases} v[1] \leftarrow c[1] \\ j \leftarrow 2 \end{cases}$$

Fai una scansione dal primo fino al penultimo termine:

Se $c[i + 1] \neq c[i]$

$$\begin{cases} v[j] \leftarrow c[i + 1] \\ j \leftarrow j + 1 \end{cases}$$

Altrimenti non fare nulla

Lunghezza del vettore $v[i]$ $L \leftarrow j - 1$

Fine**ESEMPIO****UNIFICA**

Ad ogni passo della scansione, il risultato in rosso corrisponde a un nuovo termine del vettore $v[j]$

I valori del vettore finale $v[j]$ compaiono nella penultima colonna dello schema.

i	1	2	3	4	5	6	7	$v[j]$	j	$v[j]$	j	$v[j]$	j	$v[j]$	j	$v[j]$	j
$c[i]$	1	1	2	3	4	5	5	1	1	1	1	1	1	1	1	1	1
$i = 1$	1	1							2	2	2	2	2	2	2	2	2
$i = 2$		1	2								3	3	3	3	3	3	3
$i = 3$			2	3									4	4	4	4	4
$i = 4$				3	4										5	5	5
$i = 5$					4	5											6
$i = 6$					5	5											

LUNGHEZZA: L = 5