

Capitolo 3 - Vettori (Array monodimensionali)

0. Determinazione dell'output

Indica il nuovo contenuto dell'array dopo l'esecuzione del codice riportato sulla destra, tenendo conto del valore iniziale dell'array, riportato nella figura a sinistra del codice.

a)

vet

10	20	30	40	50	60
----	----	----	----	----	----

```
for( i=0; i<6; i++ ) {  
    vet[i] = vet[ (i+1)%6 ];  
}
```

b)

vet

10	20	30	40	50	60
----	----	----	----	----	----

```
for( i=0; i<3; i++ ) {  
    vet[i] = vet[ 6 - i - 1];  
}
```

c)

vet

10	20	30	40	50	60
----	----	----	----	----	----

```
for( i=0; i<3; i++ ) {  
    temp = vet[i];  
    vet[i] = vet[ 6 - i - 1];  
    vet[6 - i - 1] = temp;  
}
```

d)

vet1

10	20	30	40	50	60
----	----	----	----	----	----

vet2

2	1	3	2	3	1
---	---	---	---	---	---

```
for( i=0; i<6; i++ ) {  
    vet1[i] = vet1[i]*vet2[i];  
}
```

e)

vet1					
10	20	30	40	50	60

vet2					
1	0	3	5	2	4

vet3					
?	?	?	?	?	?

```
for( i=0; i<6; i++ ) {  
    vet3[i] = vet1[ vet2[i] ];  
}
```

[es00_esercizi_base_cicli.pdf](#)

1. Inversione di una sequenza

Scrivi un programma che acquisisce dall'utente una sequenza di N numeri interi e, terminato l'inserimento, mostra i numeri in ordine inverso rispetto all'ordine di inserimento. Il valore di N è scelto dall'utente prima di iniziare l'inserimento, ma non può superare il valore 100.

```
Quanti numeri vuoi inserire nel vettore (max 100)? 3  
Inserisci v[0]: 5  
Inserisci v[1]: 8  
Inserisci v[2]: 2  
Output: 2, 8, 5.
```

2. Valori sopra e sotto la media

Scrivi un programma che richiede all'utente N numeri interi (N stabilito dall'utente, positivo e minore di 100). Terminato l'inserimento il programma calcola la media aritmetica fra i numeri inseriti, quindi mostra prima tutti i numeri minori o uguali alla media, e poi tutti i numeri maggiori della media.

```
Quanti numeri vuoi inserire nel vettore (max 100)? 4  
Inserisci un numero: 1  
Inserisci un numero: 8  
Inserisci un numero: 2  
Inserisci un numero: 3  
La media e' 3.5  
Minori o uguali alla media: 1, 2, 3  
Maggiori della media: 8
```

3. Numeri divisibili per 3 e per 5

Richiedere all'utente un vettore di N numeri interi e, terminato l'inserimento, visualizzare solo i numeri inseriti che sono divisibili sia per 3 che per 5.

```
Quanti numeri vuoi inserire nel vettore (max 100)? 4  
Inserisci un numero: 30  
Inserisci un numero: 10  
Inserisci un numero: 15
```

```
Inserisci un numero: 8
```

```
I numeri divisibili sia per 3 che per 5 sono: 30, 15
```

4. Prodotto scalare

Il *prodotto scalare* tra due vettori è definito come "la somma dei prodotti degli elementi corrispondenti". Esempio: vettore1 = [3,4,7,0,1] vettore2 = [1,2,0,0,5]

Il prodotto scalare è: $3*1 + 4*2 + 7*0 + 0*0 + 1*5 = 3+8+5 = 16$

Scrivi un programma che richiede all'utente due vettori di interi di ugual dimensione (N) e mostra il loro prodotto scalare.

```
Inserisci la lunghezza comune ai due vettori (max 100): 5
```

```
Inserisci il primo vettore:
```

```
v1[0] = 3
```

```
v1[1] = 4
```

```
v1[2] = 7
```

```
v1[3] = 0
```

```
v1[4] = 1
```

```
Inserisci il secondo vettore:
```

```
v2[0] = 1
```

```
v2[1] = 2
```

```
v2[2] = 0
```

```
v2[3] = 0
```

```
v2[4] = 5
```

```
il prodotto scalare e' 16;
```

5. Scambio posizioni pari/dispari

Scrivi un programma che richiede all'utente un vettore costituito da un numero pari di elementi (massimo 100 elementi). Il programma deve scambiare i valori in posizione pari con quelli in posizione dispari, e infine stampare il contenuto del vettore dopo tale operazione. Esempio di esecuzione:

```
Inserisci il numero di elementi (pari e <= 100): 8
```

```
Inserisci il vettore: 4 2 3 1 5 8 0 9
```

```
Il vettore dopo lo scambio e': 2 4 1 3 8 5 9 0
```

6. Scambio delle due meta'

Scrivi un programma che richiede all'utente un vettore costituito da un numero pari di elementi (massimo 100 elementi). Il programma deve scambiare le due meta' del vettore (cioè portare tutti gli elementi della prima metà nella seconda e viceversa) e infine stampare il contenuto del vettore dopo tale operazione. Esempio di esecuzione:

```
Inserisci il numero di elementi (pari e <= 100): 8
```

```
Inserisci il vettore: 4 2 3 1 5 8 0 9
```

```
Il vettore dopo lo scambio e': 5 8 0 9 4 2 3 1
```

7. Ricerca delle occorrenze

Scrivi un programma che, dato un vettore di N interi e un numero intero k, comunica quante volte e in quali posizioni il numero k è presente nel vettore. Il programma dovrà comportarsi come specificato dal seguente esempio.

```
Inserisci la lunghezza del vettore (max 100): 7
Inserisci i numeri: 8 5 3 5 5 2 1
Inserisci il valore da cercare (k): 5
Il valore 5 compare agli indici 1 3 4
Il valore 5 compare 3 volte
```

8. Statistica dei voti

Scrivi un programma che, dato un elenco di voti, visualizza la frequenza di occorrenza di ciascun voto. Si possono inserire solo voti interi compresi tra 2 e 10. Esempio:

```
Quanti voti vuoi inserire? 15
Inserisci i voti: 7 6 6 7 5 6 5 5 4 8 7 8 9 10 6
OUTPUT:
Voto=2 Freq=0
Voto=3 Freq=0
Voto=4 Freq=1
Voto=5 Freq=3
Voto=6 Freq=4
Voto=7 Freq=3
Voto=8 Freq=2
Voto=9 Freq=1
Voto=10 Freq=1
```

9. Funzione "trovaMinimo" su array

Realizza una funzione che riceve come argomento un array di interi e il numero di valori in esso contenuti, e restituisce il valore più piccolo tra quelli memorizzati nell'array. Prototipo della funzione: `int trovaMinimo(int vet[], int n);`

10. Unione e intersezione

Dati due insiemi A e B di numeri interi, si definisce **unione** tra A e B l'insieme composto da tutti i numeri che appartengono ad almeno uno dei due insiemi. Si definisce invece **intersezione** tra A e B l'insieme composto da tutti i numeri sia ad A che a B. Scrivi un programma che richiede all'utente due insiemi (max 100 elementi) e mostra la loro unione e la loro intersezione. Esempio:

INPUT:

A = {1,5,3,8,2}

B = {3,2,10}

OUTPUT:

Unione tra A e B = {1,5,3,8,2,10}

Intersezione tra A e B = {3,2}
