

Gli Array (Vettori)

Page • 1 link to

■ Note per prova intermedia

Domande fino a funzioni (no enum o array)

scelta multipla e v/f (logica → capire flowchart, passare da flow a codifica)

giusta +1 | non data 0 | sbagliata 0

primo e secondo blocco di slide (quelle indicate con 1 e 2 non l'azzeramento)

20 domande, 1:30 per domanda (il tempo comincia da quando si avvia il test)

Per il test finale si possono usare le slide, appunti, ecc. per questo no

Gli Array

In C++ un array è un insieme di elementi omogenei tra di loro

I singoli elementi vengono indicati con un indice (tipo le string)

C++ ▾

```
// tipo nomeArray[dimensione]
```

```
int naturali[1000];
```

```
// Per considerare un solo elemento:
```

```
naturali[5]; // prende il 6° elemento dell'array (in posizione 5)
```

Attenzione

in C++ è possibile accedere a valori non contenuti nell'array!

Questo porta a dover fare delle considerazioni in merito per evitare errori di tipo `out_of_bound` andando ad accedere a memoria non allocata.

La dichiarazione con assegnazione può avvenire in due modi:

C++ ▾

Unwrap Copy

```
// Dichiaro esplicitamente la dimensione dell'array
double temp[3] = {2,6,5};

// La dimensione viene calcolata in base al numero di elementi
int num[] = {7,5,8,3,2}
```

Gli array di char

Gli array di char sono di fatto delle stringhe

C++ ▾

```
char stringa[5] = "ciao"; // lunghezza 5 perchè è "ciao\0"
int main(){
    int i;
    for (i=0; i<6;i++)
        cout << stringa[i];
    return 0;
}
```

Passaggio di parametri

Gli array vengono SEMPRE passati per riferimento

C++ ▾

```
void function(int v[]){} // Nota: Non è necessario mettere "&"
```

il nome dell'array (v) contiene l'indirizzo del primo elemento dell'array

Stampa

Per stampare i valori di un array diventa ora essenziale usare una funzione. Questo perché bisogna accedere con un ciclo ad ogni elemento dell'array, non si può stampare tutto l'array in una volta sola.

Matrice

La matrice è un array di 2 o più dimensioni

C++ ▾

```
// tipo nomeMatrice[N][M];

int tabella[20][10]; // Crea una array da 20, ognuno con 10 valori
```

```
int cubo[10][10][10]; // Crea un cubo di valori 10x10x10
```

Array paralleli (o Associativi)

C++ ▾

```
double tempMAX[citta][anno][mese][giorno];  
double tempMIN[citta][anno][mese][giorno];
```

Questi due array (matrici) vengono gestiti in modo "sincronizzato", ad esempio:

C++ ▾

```
tempMAX["Torino"][2024][2][19];  
tempMIN["Torino"][2024][2][19];  
  
// In questo modo andiamo a prendere le temperature della stessa città  
nello stesso periodo.
```

Ovviamente è poi possibile accedere con indici diversi

C++ ▾

```
tempMAX["Torino"][2024][2][19];  
tempMAX["Milano"][2024][2][19];  
  
// In questo modo andiamo a prendere le temperature di città diverse ma  
nello stesso periodo.
```

|