



ELEMENTI DI INFORMATICA

DOCENTE: FRANCESCO MARRA

INGEGNERIA CHIMICA

INGEGNERIA ELETTRICA

SCIENZE ED INGEGNERIA DEI MATERIALI

INGEGNERIA GESTIONALE DELLA LOGISTICA E DELLA PRODUZIONE

INGEGNERIA NAVALE


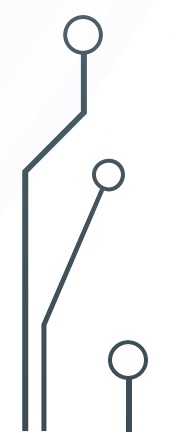
UNIVERSITÀ DEGLI STUDI DI NAPOLI FEDERICO II
SCUOLA POLITECNICA E DELLE SCIENZE DI BASE

TIPI STRUTTURATI: VETTORI E MATRICI





AGENDA

- Tipi strutturati
 - Vettori
 - matrici
- 
- 
- 

GLI ARRAY

- Un array è una struttura composta da un insieme di elementi tutti dello **stesso tipo** e con lo stesso nome
- Dimensione o *cardinalità* di un array
 - Numero di elementi dell'array
- Dichiarazione di un array

```
<Tipo> nome_array[dimensione];
```

ESEMPIO DI ARRAY

- Array di 10 elementi di tipo float, chiamato v

```
float v[10];
```



ALLOCAZIONE DELL'ARRAY

- La dimensione dell'array deve essere un valore **costante**
 - Ciò consente al compilatore di fissarne l'allocazione in memoria prima dell'esecuzione del programma
- Gli elementi vengono allocati in locazioni di memoria **consecutive**
- La dichiarazione di un array non inizializza l'array
 - Il contenuto dell'array non è definito

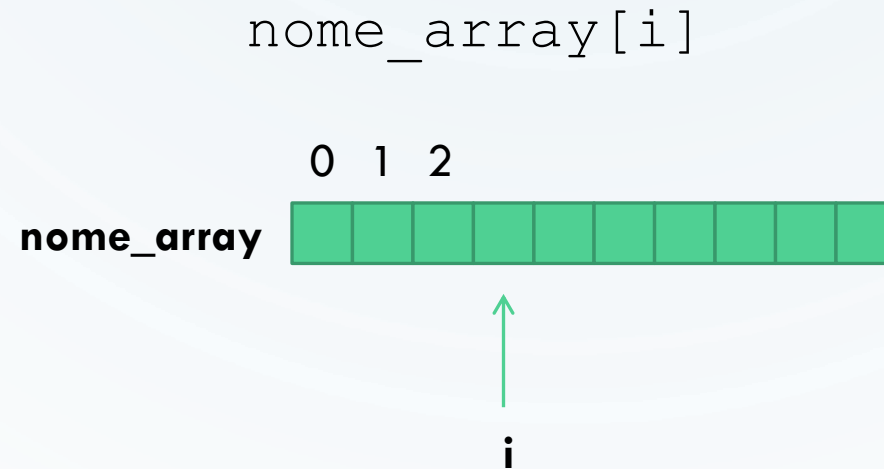
DEFINIZIONE DI UN TIPO ARRAY

- Con `typedef` si definisce un tipo array assegnando ad esso un nome
- Esempio
 - `v` è una variabile di tipo `arrayDiReali`

```
const int N=10;  
typedef float ArrayDiReali[N];  
ArrayDiReali v;
```

ACCESSO AD UN ARRAY

- La funzione di accesso ad uno degli elementi della struttura si ottiene specificandone la posizione tra parentesi quadre
 - i è detto **indice**



ACCESSO AD UN ARRAY

- Per indicare la posizione di un elemento dell'array si possono usare:

- Valori costanti \rightarrow `v[1] = 15;`

- Variabili \rightarrow `v[i] = 30;`

- Espressioni \rightarrow `v[i+1] = 45;`

ACCESSO AD UN ARRAY

- Nel caso di accesso ad un array mediante espressioni o variabili, il valore dell'indice è definito a *tempo di esecuzione*
- Nel caso si acceda a posizioni superiori a quelle dichiarate, il programma non produce messaggi di errore
 - Ma invade zone di memoria non di sua pertinenza, con conseguenze imprevedibili

```
int v[10];
```

```
v[10] = 5;
```

Da evitare



ACCESSO AD UN ARRAY

- Non esistono operazioni che coinvolgono l'array nella sua interezza, ma solo quelle definite sui singoli elementi
- Le operazioni che interessano tutti gli elementi vengono realizzati con cicli iterativi, solitamente di tipo `for`
- La dichiarazione dell'array con inizializzazione si effettua indicando i valori degli elementi tra parentesi graffe

```
float v[3] = {0.5, -13.25, 12.2};
```

ARRAY MULTIDIMENSIONALI

- Si possono anche dichiarare **array multidimensionali**
 - Sono strutture in cui sono necessarie più posizioni per identificare un elemento
 - Possono essere visti come array di array

`<Tipo> nome_array[dim1][dim2]...[dimn];`

- Esempio:
 - Array bidimensionale di interi di dimensioni 2 per 3
 - 6 elementi in totale

`int mat[2][3];`

`mat[i][j];`

mat[0][0]: prima posizione
mat[1][2]: ultima posizione

Accesso alle posizioni i e j

ARRAY MULTIDIMENSIONALI

- È possibile anche dichiarare il tipo di un array multidimensionale
 - Un array bidimensionale è anche detto *matrice*, mentre un array monodimensionale *vettore*

```
const int N=2;  
const int M=3;  
typedef int Matrice[N][M]  
Matrice mat;
```

	0	1	2
0			
1			

ALLOCAZIONE DEGLI ARRAY BIDIMENSIONALI

- Anche gli array bidimensionali vengono disposti in locazioni di memoria consecutive
 - In particolare vengono disposte le righe una dopo l'altra

```
int mat[2][3] = {1,2,3,4,5,6};
```



```
mat[0][0]=1;
```

```
mat[0][1]=2;
```

```
mat[0][2]=3;
```

```
mat[1][0]=4;
```

```
mat[1][1]=5;
```

```
mat[1][2]=6;
```

INIZIALIZZAZIONE DEGLI ARRAY BIDIMENSIONALI

- È anche possibile raggruppare le righe tra parentesi, sempre indicandole una dopo l'altra

```
int mat[2][3]    =  {{1,2,3},  
                     {4,5,6}};
```

- Se nell'inizializzazione si specificano meno valori di quelli presenti nella matrice, i valori rimanenti sono inizializzati automaticamente a ZERO
 - Ad esempio:

```
int mat[2][3] = {0};
```

CARDINALITÀ DEGLI ARRAY

- Le cardinalità fissano il numero massimo di elementi dell'array che il programma può usare
 - È possibile anche usarne una parte
 - In tal caso va gestito il grado di riempimento dell'array con una variabile che indica il numero di elementi che sono realmente usati dal programma

```
cin >> riemp;  
for (int i = 0; i < riemp; i++) {  
    cin >> v[i];  
}
```

Deve essere al massimo uguale
alla dimensione fissata per v

CARDINALITÀ DEGLI ARRAY

- Anche per gli array multidimensionali è possibile utilizzare una parte degli elementi

```
cin >> num_rig;  
cin >> num_col;  
for (int r = 0; r < num_rig; r++) {  
    for (int c = 0; c < num_col; c++) {  
        cin >> mat[r][c];  
    }  
}
```

Devono essere al massimo uguali alle dimensioni
di riga e di colonna fissate per mat


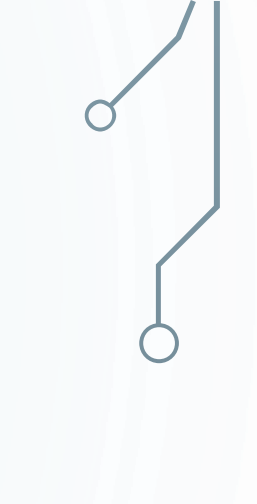
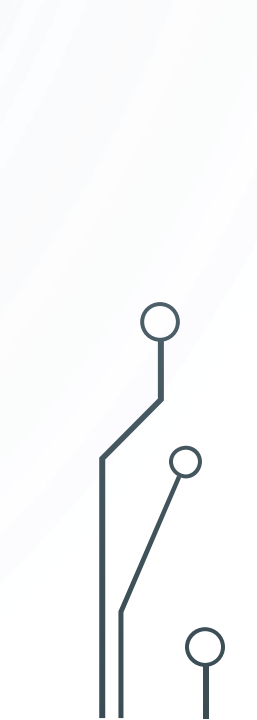
The slide features a light blue background with a subtle pattern of concentric circles. Decorative circuit-like lines with small circles at the ends are positioned along the left and right edges of the slide.

ESERCIZI

- Dato un vettore di N interi, inizializzati da tastiera, determinare il valore massimo e stamparlo a video
- Dato un vettore di N interi, inizializzati da tastiera, determinare il valore massimo e stampare a video sia il massimo che la posizione in cui esso compare



ESERCIZI RIEPILOGATIVI

- Scrivere un programma che, dato un vettore di N interi, inizializzati da tastiera, controlla se esso contiene un ulteriore valore inserito dall'utente.
- 
- 
- 

ESERCIZI RIEPILOGATIVI

- Scrivere un programma che, data una matrice $N \times M$ di numeri reali, inizializzati da tastiera, determina la media dei valori per ciascuna riga

ESERCIZI RIEPILOGATIVI

- Scrivere un programma che calcola l'integrale di una funzione polinomiale di grado n , $f(x)=a_nx^n+\dots+a_1x+a_0$, a coefficienti reali, definito tra x_{\min} e x_{\max} .
- L'utente inserisce n .
- Poi inserisce i coefficienti, che vengono assegnati agli elementi di un vettore.
- Infine, inserisce gli estremi.
- Si utilizzi la definizione di integrale come somma di aree, con $\Delta x=0.01$:
 - integrale=0
 - $x=x_{\min}+\Delta x/2$
 - Finché $x < x_{\max}$:
 - integrale aumenta di $\Delta x \cdot f(x)$
 - x aumenta di Δx

DOMANDE, DUBBI, PERPLESSITÀ

