
Linguaggio C: Allocazione dinamica di matrici

Valeria Cardellini

Corso di Calcolatori Elettronici
A.A. 2018/19

Università degli Studi di Roma "Tor Vergata"
Dipartimento di Ingegneria Civile e Ingegneria Informatica

Obiettivo

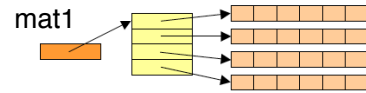
- ❑ Esaminiamo 4 possibili modi per allocare dinamica un array bidimensionale (matrice)
- ❑ Differenze
 - Matrice come blocco di memoria contiguo oppure non continuo
 - Modalità di accesso agli elementi della matrice
 - Come passare la matrice allocata dinamicamente come parametro di ingresso di una funzione

Allocazione dinamica di matrici: versione 1

- ❑ Con array di puntatori a tipo (ad es. int), blocco di memoria **non contiguo**:

//nr: numero di righe, nc: numero di colonne

```
int **mat1;  
mat1 = (int **)malloc(nr*sizeof(int *));  
for (i=0; i<nc; i++)  
    mat1[i] = malloc(nc*sizeof(int));
```

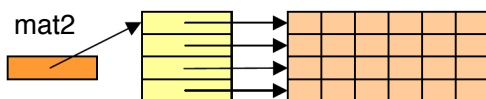


- ❑ malloc crea un array di nr puntatori a int e per ogni puntatore alloca un array di int di lunghezza nc
- ❑ Come si accede agli elementi della matrice:
`mat1[riga][colonna] = 5;`
- ❑ nr e nc possono essere variabili

Allocazione dinamica di matrici: versione 2

- ❑ Con array di puntatori a tipo (ad es. int), blocco di memoria **contiguo**:

```
int **mat2;  
mat2 = (int **)malloc(nr*sizeof(int *));  
mat2[0] = (int*)malloc(nr*nc*sizeof(int));  
for (i=1; i<nr; i++)  
    mat2[i] = mat2[0] + i*nc;
```

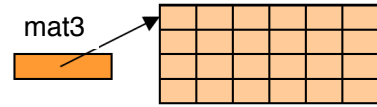


- ❑ malloc crea un array di nr puntatori, alloca un blocco per tutta la matrice, calcola e assegna ad ogni puntatore l'indirizzo di ciascuna riga
- ❑ Come si accede agli elementi della matrice:
`mat2[riga][colonna] = 5;`
- ❑ nr e nc possono essere variabili

Allocazione dinamica di matrici: versione 3

- ❑ Con array di tipo (ad es. int), blocco di memoria **contiguo** ma la matrice è simulata con un array monodimensionale:

```
int *mat3;  
mat3 = (int *)malloc(nr*nc*sizeof(int));
```

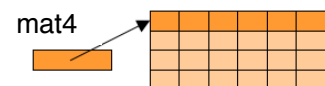


- ❑ malloc crea un blocco per tutta la matrice e accede agli elementi calcolandone la posizione (offset) riferita al primo elemento
 - La matrice è in realtà un vettore
- ❑ Accesso agli elementi della matrice:
`mat3[riga*nc+colonna] = 5;`
- ❑ nr e nc possono essere variabili

Allocazione dinamica di matrici: versione 4

- ❑ Con puntatore ad array di tipo (ad es. int), blocco di memoria **contiguo**:

```
int (*mat4)[NCOL];  
mat4 = (int (*)[NCOL])malloc(nr*sizeof(*mat4));
```



- ❑ malloc crea un blocco per tutta la matrice e lo fa puntare da un puntatore
 - Il tipo di mat4 è "puntatore a array di NCOL int"
 - *mat4 è un "vettore di NCOL int"
 - La sua dimensione è quella di una riga della matrice (NCOL)
- ❑ Accesso agli elementi della matrice:
`mat4[riga][colonna] = 5;`
- ❑ nr può essere variabile, NCOL è una costante

Passaggio di parametri

- ❑ Esaminiamo il passaggio di parametri, che avviene per riferimento
- ❑ Vedere `itineri18-19.c`