

# ALLOCAZIONE CONTIGUA E NON CONTIGUA

## ALLOCAZIONE NON-CONTIGUA:

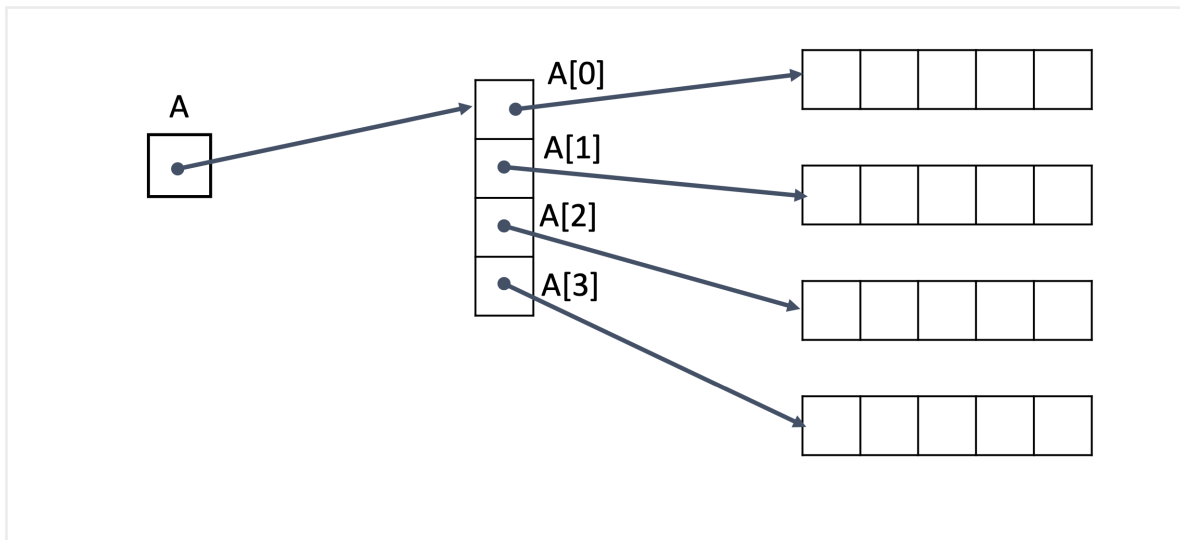
allocazione dinamica *non contigua* per matrici : abbiamo un tipo, n righe, m colonne

1. Allochiamo un'array monodimensionale di puntatori di lunghezza n

```
<tipo> ** <A> = (<tipo> **) malloc(<n> * sizeof(<tipo> *));
```

2. Inizializziamo i valori

```
for(int i=0;i < <N_R>; i++)  
    <A>[i] = (<tipo> *) malloc(<m> * sizeof(<tipo>));
```





```
1 int ** A = (int**) malloc( n * sizeof(int *) );  
2 for (int i = 0; i < n; i++)  
3     A[i] = (int *) malloc( m * sizeof(int) );
```

Accesso agli elementi rimarrà invariato.

Parametro in chiamata di funzione: come puntatore a puntatore

```
Void f(Int ** B, int n, int m);    // declare  
f(A,n,m);                          //
```

call

Rilascio della memoria:



```
1 for (int i = 0; i < n; i++)  
2     free(A[i]); // prima gli elementi  
3 free(A); // poi l'array
```

## ALLOCAZIONE CONTIGUA:

allocazione dinamica *contigua* per matrici : abbiamo un tipo, n righe, m colonne

1. Allochiamo un'array monodimensionale di puntatori di

lunghezza n

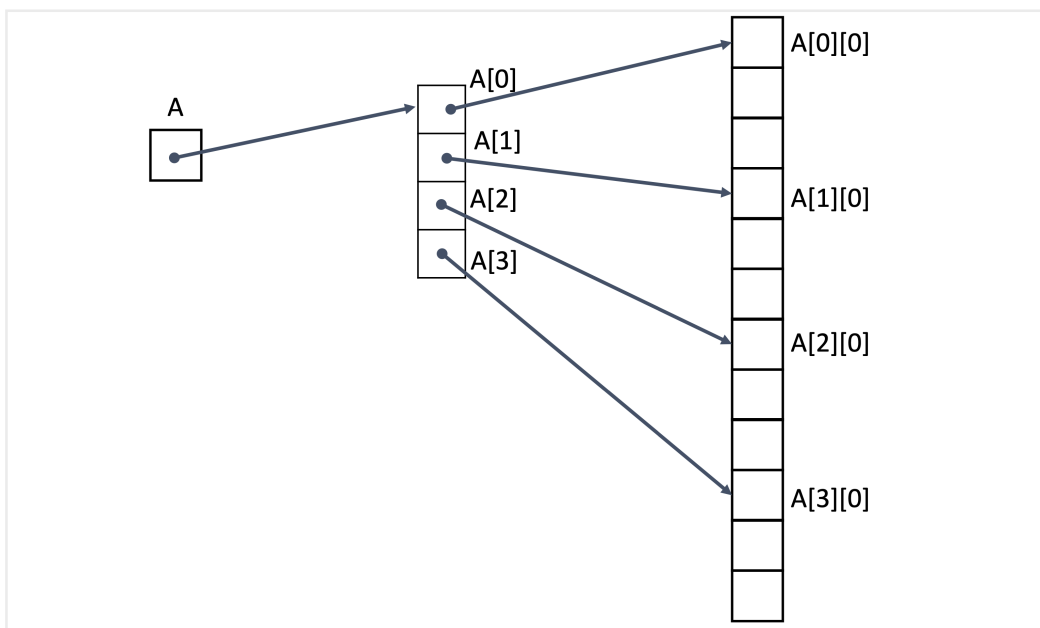
```
<tipo> ** <A> = (<tipo> **) malloc(<n> *  
sizeof(<tipo> *));
```

2. Inizializziamo il primo elemento dell'array

```
<A>[0] = (<tipo> *) malloc(<m> * <n> *  
sizeof(<tipo>));
```

3. assegnamo all'elemento i-esimo <A>[i] dell'array <A>  
con  $i \neq 0$  l'indirizzo iniziale della riga i-esima:

```
for(int i=1; i < <n>; i++)  
    <A>[i] = A[0] + i * <m>;
```



```
1  int ** A = (int**) malloc( n * sizeof(int *) );  
2  
3  A[0] = (int *) malloc( (n * m) * sizeof(int));  
4  
5  for (int i = 1; i < n; i++)  
6      A[i] = A[0] + (i * m);
```

Accesso agli elementi rimarrà invariato.

Parametro in chiamata di funzione: come puntatore a

puntatore

```
Void f(Int ** B, int n, int m);    // declare  
f(A,n,m);                        //
```

call

Rilascio della memoria:



```
1  free(A[0]); // prima l'elemento zero  
2  free(A);   // poi l'array
```

*Nota: Una variabile matrice A NON allocata dinamicamente NON può essere passata come argomento in un chiamata di funzione il cui corrispondente parametro formale è un puntatore di puntatore al tipo degli elementi della matrice.*

*Per consentire il passaggio come puntatore di puntatore è necessario definire un array di puntatori di lunghezza pari al numero di righe di A e inizializzare l'elemento i-esimo dell'array all'indirizzo iniziale A[i] della riga i-esima.*