Dichiarazione, creazione e gestione di array in Java

Array in Java - creazione

- La creazione fa una inizializzazione implicita:

  ```
  num = new int[10];
  ```

  con valore `0` per `int` e `double`, `false` per i `boolean`.

- Dopo la creazione, un array ha lunghezza fissata (e non può cambiare nel programma tramite una nuova dichiarazione).

- Gli elementi di un array possono essere assegnati ad elementi di un array dello stesso tipo:

```
int x[] = new int[10];
int y[] = new int[50];

x[3] = y[5];
```

Array in Java - copia

- Se si vuole copiare un array in un altro:

```
int vet1[] = new int[20];
int vet2[] = new int[20];

for(int i=0; i< 20; i++)
    vet1[i] = vet2[i];
```

- Se si vuole copiare una parte di array in un altro:

```
int vet3[] = new int[10];

for(int i=0; i< 10; i++)
    vet3[i] = vet1[i];
```

Per copiare la prima metà di `vet1` in `vet3`.

Array in Java - copia

- Se si vuole copiare una parte di array in un altro:

```
int vetA[] = new int[20];
int vetB[] = new int[10];
```

- Per copiare la seconda metà di **vetA** in **vetB**.

```
for(int i=0; i< 10; i++)
    vetB[i] = vetA[i+10];
```

- oppure

```
for(int i=10; i< 20; i++)
    vetB[i-10] = vetA[i];
```

Array in Java - copia

- Se si vuole copiare gli elementi di **`vet1`** in **`vet2`** in ordine inverso:

```
int vet1[] = new int[20];
int vet2[] = new int[20];

for(int i=0; i< 20; i++)
   vet2[i] = vet1[19-i];
```

Array in Java - copia

- Se si vuole copiare da array ad array si può usare il metodo

```
System.arraycopy(arr-orig, pos, arr-des, pos, cont)
```

Esempi:

```
System.arraycopy(vet1, 0, vet2, 0, 15);
```

Copia i primi 15 elementi di vet1 in vet2

```
System.arraycopy(vet2, 5, vet3, 0, 10)
```

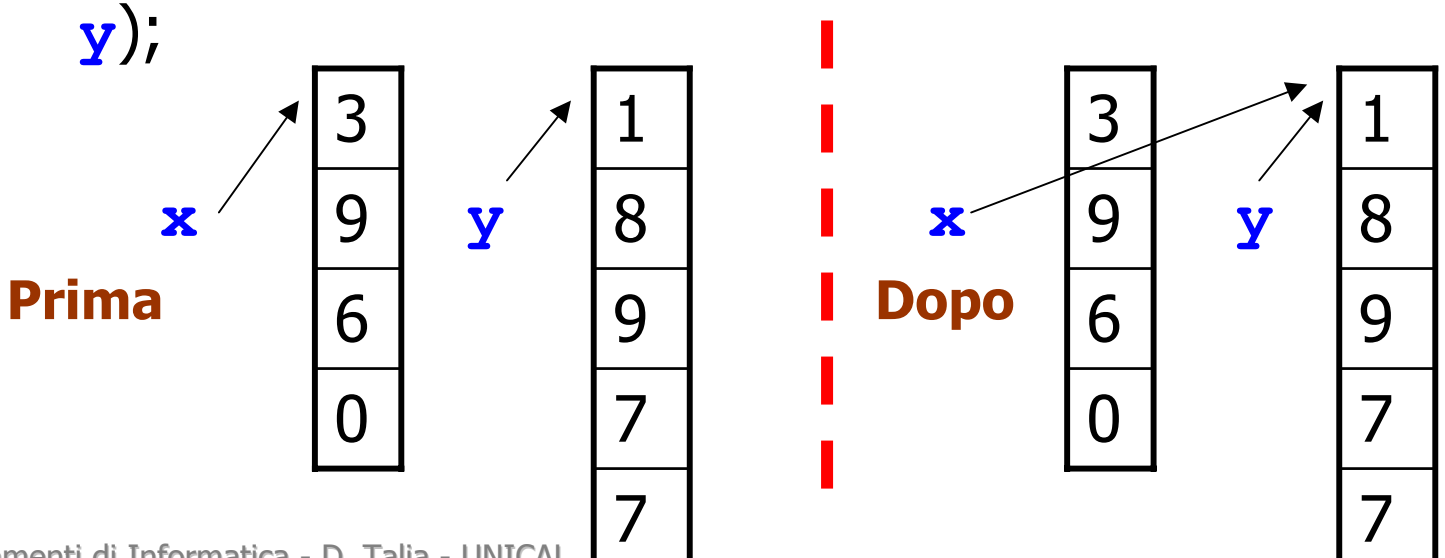
Copia 10 elementi (a partire dal sesto) di vet2 in vet3.

Array in Java – assegnamento tra array

- Si possono assegnare anche array (ma bisogna porre attenzione a quello che accade se i due array non hanno lunghezza uguale **!**)

```
x = y;
```

- Dopo questa operazione le due variabili fanno riferimento alla stessa locazione di memoria (quella di `y`);



Array in Java - lunghezza

- Ad ogni variabile array è associata implicitamente una variabile istanza **`length`**:

  `int [] seq = new int[5]`

  `seq.length` avrà valore 5

- Esempi di uso
  - **`N = seq.length;`** assegna 5 ad N
  - **`System.out.println(seq.length);`**

    Visualizza il valore 5
- Uso nel ciclo for
  - **`for(int i=0; i<vet1.length; i++)`**

    `vet1[i] = vet2[i];`

Array in Java – Massimo tra due numeri

```
class maxtraduenum
{ public static void main(String args[])
  { int[] x;
    int max;

 x = new[2];
 x[0]= Console.readInt("dammi il primo numero");
 x[1]=Console.readInt("dammi il secondo numero");
 if (x[0] > x[1])
   max=x[0];
 else
  max=x[1];
 System.out.println ("Massimo = " + max);
 }
}
```

Array in Java – Massimo tra N numeri

```
class cercamassimo
{
 public static void main(String args[])
  {
  int[] seq;
  int max, ind;

  seq = new int[10];
  for (ind=0; ind < 10; ind++)
  {
   seq[ind] = Console.readInt("dammi un numero");
  }
  max = seq[0];
  for(ind=0; ind < 10; ind++)
  {
   if (seq[ind] > max)
     max = seq[ind];
  }
  System.out.println ("Massimo = " + max);
  }
}
```

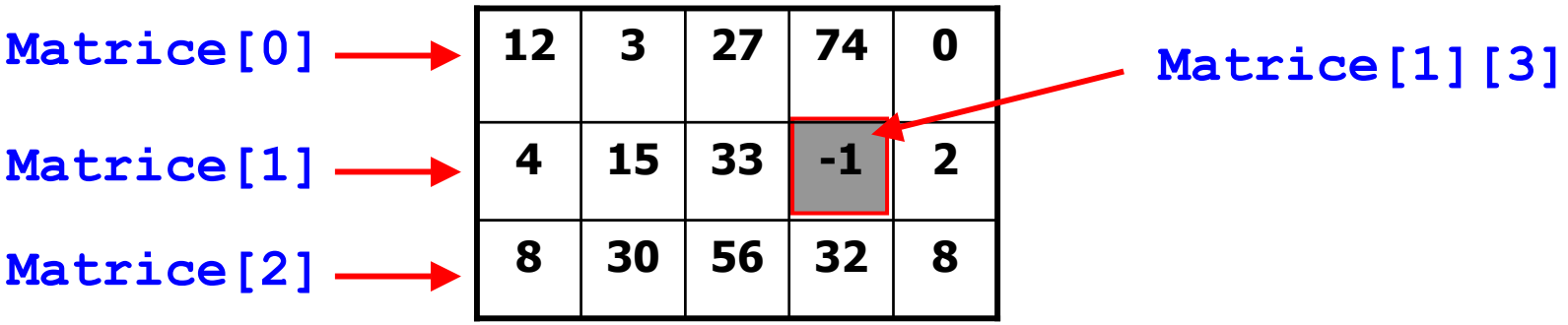
Array Multidimensionali

Array bidimensionali

- In Java si possono avere array a più dimensioni.
- Gli array a due dimensioni in Java si realizzano come array di array.
- Dichiarazione:

```
int [][] Matrice = new int[3][5];
```

[riga, colonna]



Array Multidimensionali

Array bidimensionali

- Assegnamento di un elemento

```
matrice[3][5] = 10;
```

- Assegnamento di una riga

```
matrice[3] = {10, 15, 20, 25, 30, 35};
```

- Il numero di righe è dato da

```
int nrighe = matrice.length
```

- Il numero di colonne è dato da

```
int ncolonne = matrice[0].length
```

Array con righe di lunghezza variabile

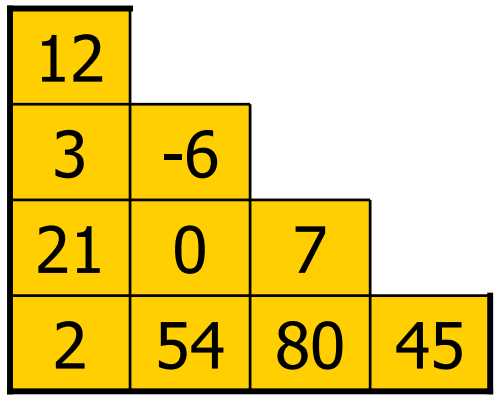
- In un array bidimensionale in Java si possono avere righe di lunghezza differente.

- Dichiarazione:

```
int[][] tabella = new int[4][];
```

- Assegnamento

```
for(int i=0; i<tabella.length; i++)
    tabella[i] = new int[i+1];
```



Array – Leggere i valori di una matrice

```
class matrice
{
  . . . . .

public void leggimatrice()
{
   int[][] mat;
   int i,j;

   mat = new int[4][5];

 for (i=0; i < 4; i++)
 {
  for(j=0; j < 5; j++)
    mat[i][j]= Console.readInt("dammi un elemento");
 }
   . . . .
}
```