

Titolo unità didattica: Strutture dati: array

[07]

Titolo modulo : Algoritmi di base su array 2D

[06-T]

Sviluppo di algoritmi per problemi di base con dati in Array 2D

Argomenti trattati:

- ✓ array 2D
- ✓ algoritmo incrementale per il massimo, minimo di array 2D
- ✓ algoritmo di somma, media di array 2D
- ✓ accesso a porzioni di array 2D (diagonale, triangolo,..)

Prerequisiti richiesti: AP-07-01-T

problema:

calcolo del massimo dei valori di un array 2D

dati di input: l'array 2D (variabile **a**), il numero di righe dell'array (variabile **n**) il numero di colonne dell'array (variabile **m**)

dato di output: il valore massimo (variabile **max_2D**)

costrutto ripetitivo: 2 cicli **for** **innestati**

operazione ripetuta (al generico passo **i,j**):

confrontare l'(**i,j**)-simo elemento

dell'array 2D con il massimo dei precedenti valori dell'array e determinare il nuovo massimo

problema:

calcolo del massimo dei valori di un array 2D

36	21	44	11
18	49	27	13
22	25	10	16
52	35	59	67



max_2D

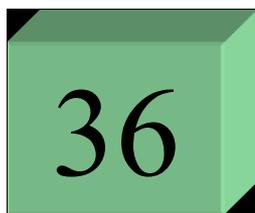
inizializzazione

```
for (i=0;i<n;i++) {  
  for (j=0;j<m;j++) {  
    confronto con  
    l'elemento [i][j]  
  }  
}
```

problema:

calcolo del massimo dei valori di un array 2D

36	21	44	11
18	49	27	13
22	25	10	16
52	35	59	67



max_2D

inizializzazione

```
for (i=0;i<n;i++) {  
  for (j=0;j<m;j++) {  
    confronto con  
    l'elemento [i][j]  
  }  
}
```

problema:

calcolo del massimo dei valori di un array 2D

36	21	44	11
18	49	27	13
22	25	10	16
52	35	59	67



max_2D

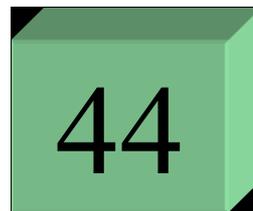
inizializzazione

```
for (i=0;i<n;i++) {  
  for (j=0;j<m;j++) {  
    confronto con  
    l'elemento [i][j]  
  }  
}
```

problema:

calcolo del massimo dei valori di un array 2D

36	21	44	11
18	49	27	13
22	25	10	16
52	35	59	67



max_2D

inizializzazione

```
for (i=0;i<n;i++) {  
    for (j=0;j<m;j++) {  
        confronto con  
        l'elemento [i][j]  
    }  
}
```

problema:

calcolo del massimo dei valori di un array 2D

36	21	44	11
18	49	27	13
22	25	10	16
52	35	59	67



max_2D

inizializzazione

```
for (i=0;i<n;i++) {  
  for (j=0;j<m;j++) {  
    confronto con  
    l'elemento [i][j]  
  }  
}
```

problema:

calcolo del massimo dei valori di un array 2D

36	21	44	11
18	49	27	13
22	25	10	16
52	35	59	67



max_2D

inizializzazione

```
for (i=0;i<n;i++) {  
  for (j=0;j<m;j++) {  
    confronto con  
    l'elemento [i][j]  
  }  
}
```

e così via...

inizializzazione:

```
max_2D = a[0][0] ;
```

costrutto di selezione all'interno dei due cicli innestati:

```
if (a[i][j] > max_2D)  
    { max_2D = a[i][j]; }
```

```
float massimo_a2D(float a[][], int n, int m) {
    int i,j;
    float max_2D;
    max_2D = a[0][0] ;
    for (i=0; i < n; i++) {
        for (j=0; j < m; j++) {
            if (a[i][j] > max_2D)
                { max_2D = a[i][j]; }
        }
    }
    return max_2D ;
}
```

$n * m$
confronti

ATTENZIONE: da modificare in C

problema:

calcolo del minimo dei valori di un array 2D e dei suoi indici

```
void minimo_a2D_ind(in: float a[][], int n, int m;  
                   out: float min_2D, int i_min, int j_min) {  
    int i, j;  
    min_2D = a[0][0]; i_min = 0; j_min = 0 ;  
    for (i=0; i < n; i++) {  
        for (j=0; j < m; j++) {  
            if (a[i][j] < min_2D)  
            { min_2D = a[i][j] ;  
              i_min = i ; j_min = j;  
            }  
        }  
    }  
}
```

$n*m$
confronti

ATTENZIONE: da modificare in C

problema:

calcolo della media dei valori di un array 2D

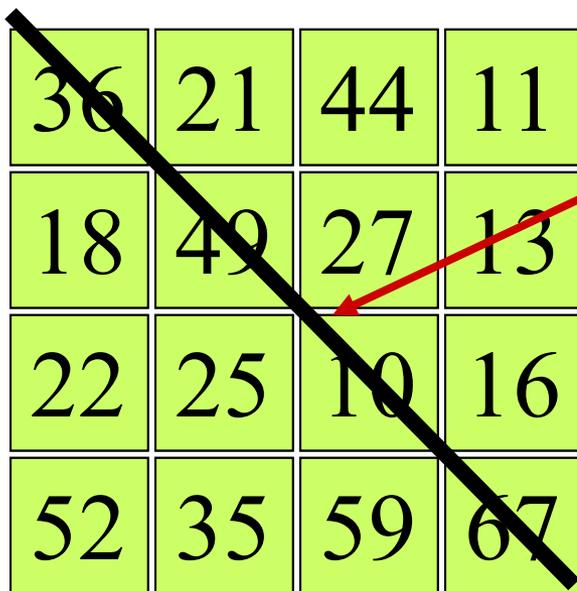
```
float media_a2D(float a[][], int n, int m; ) {  
    int i, j;  
    float somma ;  
    somma = 0.0 ;  
    for (i=0; i < n; i++) {  
        for (j=0; j < m; j++) {  
            somma = somma + a[i][j] ;  
        }  
    }  
    return somma/(float)(n*m) ;  
}
```

$n*m$
somme

ATTENZIONE: da modificare in C

problema:

calcolo del massimo dei valori della **diagonale**
(principale) di un array 2D



36	21	44	11
18	49	27	13
22	25	10	16
52	35	59	67

diagonale principale

problema:

calcolo del massimo dei valori della **diagonale**
(principale) di un array 2D

36	21	44	11
18	49	27	13
22	25	10	16
52	35	59	67



indici

problema:

calcolo del massimo dei valori della **diagonale**
(principale) di un array 2D

36	21	44	11
18	49	27	13
22	25	10	16
52	35	59	67



indici

problema:

calcolo del massimo dei valori della **diagonale**
(principale) di un array 2D

36	21	44	11
18	49	27	13
22	25	10	16
52	35	59	67



indici

problema:

calcolo del massimo dei valori della **diagonale**
(principale) di un array 2D

36	21	44	11
18	49	27	13
22	25	10	16
52	35	59	67



indici

problema:

calcolo del massimo dei valori della **diagonale**
(principale) di un array 2D

36	21	44	11
18	49	27	13
22	25	10	16
52	35	59	67

36

max_diag

```
inizializzazione  
for (i=1; i<n; i++) {  
    confronto con  
    l'elemento [i][i]  
}
```

un solo ciclo **for**

problema:

calcolo del massimo dei valori della **diagonale**
(principale) di un array 2D

36	21	44	11
18	49	27	13
22	25	10	16
52	35	59	67

36

max_diag

inizializzazione

```
for (i=1; i<n; i++) {  
    confronto con  
    l'elemento [i][i]  
}
```

un solo ciclo **for**

problema:

calcolo del massimo dei valori della **diagonale**
(principale) di un array 2D

36	21	44	11
18	49	27	13
22	25	10	16
52	35	59	67

49

max_diag

inizializzazione

```
for (i=1; i<n; i++) {  
    confronto con  
    l'elemento [i][i]  
}
```

un solo ciclo **for**

problema:

calcolo del massimo dei valori della **diagonale**
(principale) di un array 2D

36	21	44	11
18	49	27	13
22	25	10	16
52	35	59	67

49

max_diag

inizializzazione

```
for (i=1; i<n; i++) {  
    confronto con  
    l'elemento [i][i]  
}
```

un solo ciclo **for**

problema:

calcolo del massimo dei valori della **diagonale**
(principale) di un array 2D

36	21	44	11
18	49	27	13
22	25	10	16
52	35	59	67

49

max_diag

inizializzazione

```
for (i=1; i<n; i++) {  
    confronto con  
    l'elemento [i][i]  
}
```

un solo ciclo **for**

problema:

calcolo del massimo dei valori della **diagonale**
(principale) di un array 2D

36	21	44	11
18	49	27	13
22	25	10	16
52	35	59	67

67

max_diag

n
confronti

inizializzazione
for (i=1; i<n; i++) {
 confronto con
 l'elemento [i][i]
}

un solo ciclo **for**

problema:

calcolo del massimo dei valori del **triangolo superiore** di un array 2D

triangolo
superiore

36	21	44	11
18	49	27	13
22	25	10	16
52	35	59	67

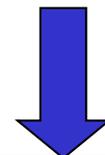
problema:

calcolo del massimo dei valori del **triangolo superiore** di un array 2D

36	21	44	11
18	49	27	13
22	25	10	16
52	35	59	67



indici



indice di riga **minore o uguale**
dell'indice di colonna

problema:

calcolo del massimo dei valori del **triangolo superiore** di un array 2D

36	21	44	11
18	49	27	13
22	25	10	16
52	35	59	67



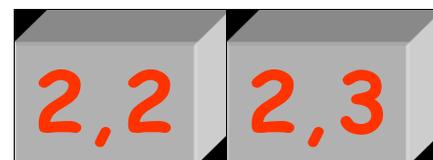
indici

indice di riga **minore o uguale**
dell'indice di colonna

problema:

calcolo del massimo dei valori del **triangolo superiore** di un array 2D

36	21	44	11
18	49	27	13
22	25	10	16
52	35	59	67



indici

indice di riga **minore o uguale**
dell'indice di colonna

problema:

calcolo del massimo dei valori del **triangolo superiore** di un array 2D

36	21	44	11
18	49	27	13
22	25	10	16
52	35	59	67



indici

indice di riga **minore o uguale**
dell'indice di colonna

problema:

calcolo del massimo dei valori del **triangolo superiore** di un array 2D

36	21	44	11
18	49	27	13
22	25	10	16
52	35	59	67



max_trisup

inizializzazione

```
for (i=0; i<n; i++) {  
    for (j=i; j<m; j++) {  
        confronto con  
        l'elemento [i][j]  
    }  
}
```

problema:

calcolo del massimo dei valori del **triangolo superiore** di un array 2D

36	21	44	11
18	49	27	13
22	25	10	16
52	35	59	67

36

max_trisup

inizializzazione

```
for (i=0; i<n; i++) {  
    for (j=i; j<m; j++) {  
        confronto con  
        l'elemento [i][j]  
    }  
}
```

problema:

calcolo del massimo dei valori del **triangolo superiore** di un array 2D

36	21	44	11
18	49	27	13
22	25	10	16
52	35	59	67

36

max_trisup

inizializzazione

```
for (i=0; i<n; i++) {  
    for (j=i; j<m; j++) {  
        confronto con  
        l'elemento [i][j]  
    }  
}
```

problema:

calcolo del massimo dei valori del **triangolo superiore** di un array 2D

36	21	44	11
18	49	27	13
22	25	10	16
52	35	59	67

44

max_trisup

inizializzazione

```
for (i=0; i<n; i++) {  
    for (j=i; j<m; j++) {  
        confronto con  
        l'elemento [i][j]  
    }  
}
```

problema:

calcolo del massimo dei valori del **triangolo superiore** di un array 2D

36	21	44	11
18	49	27	13
22	25	10	16
52	35	59	67

44

max_trisup

inizializzazione

```
for (i=0; i<n; i++) {  
    for (j=i; j<m; j++) {  
        confronto con  
        l'elemento [i][j]  
    }  
}
```

problema:

calcolo del massimo dei valori del **triangolo superiore** di un array 2D

36	21	44	11
18	49	27	13
22	25	10	16
52	35	59	67

44

max_trisup

inizializzazione

```
for (i=0; i<n; i++) {  
    for (j=i; j<m; j++) {  
        confronto con  
        l'elemento [i][j]  
    }  
}
```

problema:

calcolo del massimo dei valori del **triangolo superiore** di un array 2D

36	21	44	11
18	49	27	13
22	25	10	16
52	35	59	67

49

max_trisup

inizializzazione

```
for (i=0; i<n; i++) {  
    for (j=i; j<m; j++) {  
        confronto con  
        l'elemento [i][j]  
    }  
}
```

problema:

calcolo del massimo dei valori del **triangolo superiore** di un array 2D

36	21	44	11
18	49	27	13
22	25	10	16
52	35	59	67

49

max_trisup

inizializzazione

```
for (i=0; i<n; i++) {  
    for (j=i; j<m; j++) {  
        confronto con  
        l'elemento [i][j]  
    }  
}
```

problema:

calcolo del massimo dei valori del **triangolo superiore** di un array 2D

36	21	44	11
18	49	27	13
22	25	10	16
52	35	59	67

49

max_trisup

inizializzazione

```
for (i=0; i<n; i++) {  
    for (j=i; j<m; j++) {  
        confronto con  
        l'elemento [i][j]  
    }  
}
```

problema:

calcolo del massimo dei valori del **triangolo superiore** di un array 2D

36	21	44	11
18	49	27	13
22	25	10	16
52	35	59	67

49

max_trisup

inizializzazione

```
for (i=0; i<n; i++) {  
    for (j=i; j<m; j++) {  
        confronto con  
        l'elemento [i][j]  
    }  
}
```

problema:

calcolo del massimo dei valori del **triangolo superiore** di un array 2D

36	21	44	11
18	49	27	13
22	25	10	16
52	35	59	67

49

max_trisup

inizializzazione

```
for (i=0; i<n; i++) {  
    for (j=i; j<m; j++) {  
        confronto con  
        l'elemento [i][j]  
    }  
}
```

problema:

calcolo del massimo dei valori del **triangolo superiore** di un array 2D

36	21	44	11
18	49	27	13
22	25	10	16
52	35	59	67

49

max_trisup

inizializzazione

```
for (i=0; i<n; i++) {  
    for (j=i; j<m; j++) {  
        confronto con  
        l'elemento [i][j]  
    }  
}
```

problema:

calcolo del massimo dei valori del **triangolo superiore** di un array 2D

36	21	44	11
18	49	27	13
22	25	10	16
52	35	59	67

67

max_trisup

inizializzazione

```
for (i=0; i<n; i++) {  
    for (j=i; j<m; j++) {  
        confronto con  
        l'elemento [i][j]  
    }  
}
```

4 + 3 + 2 + 1

confronti

problema:

calcolo del massimo dei valori del **triangolo superiore** di un array 2D

36	21	44	11
18	49	27	13
22	25	10	16
52	35	59	67

67

max_trisup

inizializzazione

```
for (i=0; i<n; i++) {  
    for (j=i; j<m; j++) {  
        confronto con  
        l'elemento [i][j]  
    }  
}
```

$n*(n+1)/2$
confronti