

Πίνακες

Μεθοδολογία (I)

ΒΑΣΙΚΗ ΕΠΕΞΕΡΓΑΣΙΑ (*int A[4][5];*)

ΕΙΣΟΔΟΣ κατά Γραμμές	ΕΙΣΟΔΟΣ κατά Στήλες
<pre>for (i=0; i<4; i++) { for (j=0; j<5; j++) { printf("Give value:"); scanf("%i", &A[i][j]); } }</pre>	<pre>for (j=0; j<5; j++) { for (i=0; i<4; i++) { printf("Give value:"); scanf("%i", &A[i][j]); } }</pre>
ΕΞΟΔΟΣ κατά Γραμμές	ΕΞΟΔΟΣ κατά Στήλες
<pre>for (i=0; i<4; i++) { for (j=0; j<5; j++) { printf("%i ", A[i][j]); } printf("\n"); }</pre>	<pre>for (j=0; j<5; j++) { for (i=0; i<4; i++) { printf("%i ", A[i][j]); } printf("\n"); }</pre>

Μεθοδολογία (II)

ΒΑΣΙΚΗ ΕΠΕΞΕΡΓΑΣΙΑ (*int A[4][5];*)

ΑΡΧΙΚΟΠΟΙΗΣΗ σε σταθερά c (π.χ. c = 0)	ΑΡΧΙΚΟΠΟΙΗΣΗ
<pre>for (i=0; i<4; i++) { for (j=0; j<5; j++) { A[i][j] = 0; } }</pre>	<pre>for (j=0; j<5; j++) { for (i=0; i<4; i++) { if (i<j) A[i][j] = 0; else if (i == j) A[i][j] = 1; else A[i][j] = 2; } }</pre>

Μεθοδολογία (III)

ΒΑΣΙΚΗ ΕΠΕΞΕΡΓΑΣΙΑ (*int A[4][5];*)

ΑΘΡΟΙΣΜΑ , ΜΕΣΟΣ ΟΡΟΣ κατά γραμμές	ΑΘΡΟΙΣΜΑ , ΜΕΣΟΣ ΟΡΟΣ συνολικά
<pre>for (i=0; i<4; i++) { sr[i] = 0; for (j=0; j<5; j++) { sr[i] += A[i][j]; } mr[i] = sr[i] / 5; }</pre>	<pre>s = 0; for (i=0; i<4; i++) { for (j=0; j<5; j++) { s += A[i][j]; } } m = s/(4*5);</pre>
ΑΘΡΟΙΣΜΑ , ΜΕΣΟΣ ΟΡΟΣ κατά στήλες	
<pre>for (j=0; j<5; j++) { sc[j] = 0; for (i=0; i<4; i++) { sc[j] += A[i][j]; } mc[j] = sc[j] / 4; }</pre>	

Μεθοδολογία (IV)

ΜΕΓΙΣΤΟ & Θέση κατά γραμμές	ΜΕΓΙΣΤΟ & Θέση συνολικά
<pre>for (i=0; i<4; i++) { maxr[i] = A[i][0]; maxrp[i] = 0; for (j=0; j<5; j++) { if (A[i][j] > maxr[i]) { maxr[i] = A[i][j]; maxrp[i] = j; } } }</pre>	<pre>max = A[0][0]; maxpr = 0; maxpc = 0; for (i=0; i<4; i++) { for (j=0; j<5; j++) { if (A[i][j] > max) { max = A[i][j]; maxpr = i; maxpc = j; } } }</pre>
ΜΕΓΙΣΤΟ & Θέση κατά στήλες	
<pre>for (j=0; j<5; j++) { maxc[j] = A[0][j]; maxcp[j] = 0; for (i=0; i<4; i++) { if (A[i][j] > maxc[j]) { maxc[j] = A[i][j]; maxcp[j] = i; } } }</pre>	

Μεθοδολογία (V)

ΕΛΑΧΙΣΤΟ & Θέση κατά γραμμές	ΕΛΑΧΙΣΤΟ & Θέση συνολικά
<pre>for (i=0; i<4; i++) { minr[i] = A[i][0]; minrp[i] = 0; for (j=0; j<5; j++) { if (A[i][j] < minr[i]) { minr[i] = A[i][j]; minrp[i] = j; } } }</pre>	<pre>min = A[0][0]; minpr = 0; minpc = 0; for (i=0; i<4; i++) { for (j=0; j<5; j++) { if (A[i][j] < min) { min = A[i][j]; minpr = i; minpc = j; } } }</pre>
ΕΛΑΧΙΣΤΟ & Θέση κατά στήλες	
<pre>for (j=0; j<5; j++) { minc[j] = A[0][j]; mincp[j] = 0; for (i=0; i<4; i++) { if (A[i][j] < minc[j]) { minc[j] = A[i][j]; mincp[j] = i; } } }</pre>	

Μεθοδολογία (VI)

ΤΑΞΙΝΟΜΗΣΗ κατά γραμμές

```
for (k=0; k<4; k++)  
{  
    for (i=1; i<5; i++)  
    {  
        for (j=4; j>=i; j--)  
        {  
            if (A[k][j-1] > A[k][j])  
            {  
                temp = A[k][j-1];  
                A[k][j-1] = A[k][j];  
                A[k][j] = temp;  
            }  
        }  
    }  
}
```

ΤΑΞΙΝΟΜΗΣΗ κατά στήλες

```
for (k=0; k<5; k++)  
{  
    for (i=1; i<4; i++)  
    {  
        for (j=3; j>=i; j--)  
        {  
            if (A[j-1][k] > A[j][k])  
            {  
                temp = A[j-1][k];  
                A[j-1][k] = A[j][k];  
                A[j][k] = temp;  
            }  
        }  
    }  
}
```

Ασκήσεις (I)

Να σχεδιάσετε την μορφή που θα έχει ο πίνακας μετά την εκτέλεση του ακόλουθου τμήματος αλγορίθμου

```
int A[5][5] = {0};  
for (i=1; i<=3; i++)  
{  
    for (j=1; j<=3; j++)  
    {  
        A[i-1][j-1] = 1;  
        A[i][j-1] = 1;  
        A[i+1][j+1] = 1;  
        A[i][j+1] = 1;  
    }  
}
```

1	1	1	0	0
1	1	1	1	1
1	1	1	1	1
1	1	1	1	1
0	0	1	1	1

Ασκήσεις (II)

Να σχεδιάσετε την μορφή που θα έχει ο πίνακας μετά την εκτέλεση του ακόλουθου τμήματος αλγορίθμου

```
int A[5][5] = {0};  
for (i=0; i<5; i++)  
{  
    A[i][i] = 1;  
    A[i][4-i] = 1;  
}
```

1	0	0	0	1
0	1	0	1	0
0	0	1	0	0
0	1	0	1	0
1	0	0	0	1

Ασκήσεις (III)

Να σχεδιάσετε την μορφή που θα έχει ο πίνακας μετά την εκτέλεση του ακόλουθου τμήματος αλγορίθμου

```
char A[9][9] = {' '};  
for (i=0; i<=3; i++)  
{  
    for (j=4-i; j<=4+i; j++)  
    {  
        A[i][j] = '*';  
        A[8-i][j] = '*';  
    }  
}  
for (j=0; j<9; j++)  
{  
    A[4][j] = '*';  
}
```

```
      *  
    * * *  
  * * * * *  
* * * * * * *  
* * * * * * *  
  * * * * *  
    * * * * *  
      * * *  
        *
```

Ασκήσεις (IV)

Να σχεδιάσετε την μορφή που θα έχει ο πίνακας μετά την εκτέλεση του ακόλουθου τμήματος αλγορίθμου

```
char A[9][9] = {' '};  
for (i=0; i<=4; i++)  
{  
    for (j=4-i; j<=4+i; j++)  
    {  
        A[i][j] = '*';  
    }  
}  
for (i=5; i<9; i++)  
{  
    for (j=3; j<=5; j++)  
    {  
        A[i][j] = '*';  
    }  
}
```

```
      *  
    * * *  
  * * * * *  
* * * * * * *  
* * * * * * * * *  
  * * *  
    * * *  
      * * *  
        * * *
```

Μελέτη Άσκησης

Σε έναν αγώνα άρσης βαρών συμμετάσχουν 8 αθλητές, και διαγωνίζονται σε δύο κινήσεις: αρασέ και ζετέ. Κάθε αθλητής κάνει 5 προσπάθειες σε κάθε κίνηση. Το αποτέλεσμα κάθε αθλητή σε κάθε κίνηση είναι το μέγιστο αποτέλεσμα από τις 5 προσπάθειες, ενώ το τελικό αποτέλεσμα είναι το άθροισμα των δύο καλύτερων αποτελεσμάτων. Να γραφεί πρόγραμμα το οποίο:

- Να ορίζει ένα πίνακα με τον αριθμό κάθε αθλητή (ακέραιος τριψήφιος).
- Να ορίζει 2 πίνακες AR και ZE που θα περιέχουν τα αποτελέσματα από όλες τις προσπάθειες κάθε αθλητή.
- Να υπολογίζει την καλύτερο αποτέλεσμα σε κάθε κίνηση και σε ποια προσπάθεια πραγματοποιήθηκε.
- Να υπολογίζει το σύνολο για κάθε αθλητή.
- Να εμφανίζει τα αποτελέσματα σε φθίνουσα σειρά ως προς το σύνολο:

Αθλητής: 345 Αρασέ: 145(3) Ζετέ: 160(1) Σύνολο: 305
Αθλητής: 288 Αρασέ: 145(2) Ζετέ: 155(5) Σύνολο: 300

...