



**Τμήμα Μηχανικών Η/Υ & Πληροφορικής
Πανεπιστήμιο Πατρών**

Εισαγωγή στον Προγραμματισμό

Η γλώσσα προγραμματισμού C

Με μια ματιά

- Τύπος Πίνακα
- Μεταβλητές με ενδείκτη
- Αλφαριθμητικά - Πίνακες
- Δισδιάστατος – Πολυδιάστατος Πίνακας

Τύπος Πίνακα

- Σύνθετος τύπος δεδομένων
- Αναπαριστά ένα σύνολο ομοειδών τιμών
 - π.χ. βαθμοί ενός μαθητή, θερμοκρασίες ενός μήνα
- Χαρακτηριστικά: μέγεθος πίνακα, τύπος στοιχείων

Δήλωση - Αρχικοποίηση

□ Δήλωση

- <τύπος> <όνομα-πίνακα> [<μέγεθος>]
- `float grades[5];`

□ Αρχικοποίηση

- `float grades[5]={10, 8, 12.5, 15.5, 14};`
- `float grades[]={10, 8, 12.5};`

□ Global και static arrays

- Αρχικοποιούνται αυτόματα με 0 (εκτός αν έχετε δηλώσει explicit αρχικοποίηση)

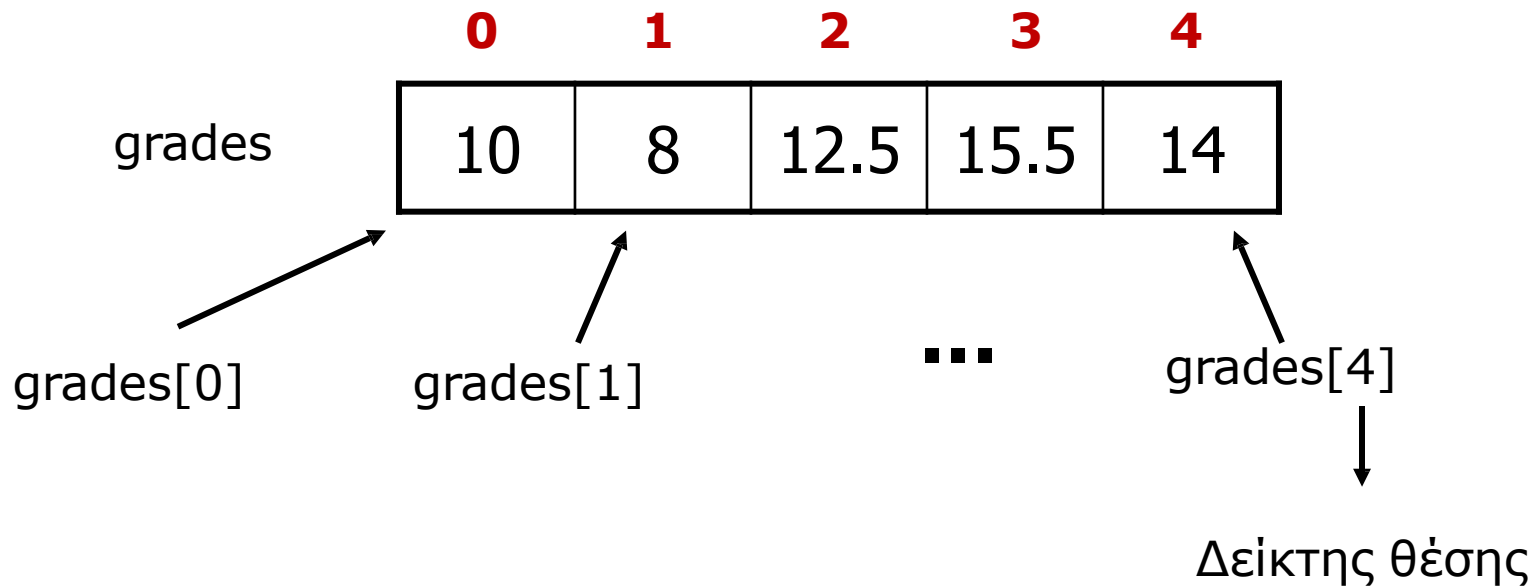
□ Local arrays

- ΔΕΝ αρχικοποιούνται by default
- Αν αρχικοποιήσετε μερικές τιμές, οι υπόλοιπες αρχικοποιούνται με 0
- Συνεπώς, για αρχικοποίηση όλων με 0

- `int myArray[1024] = {0};`

Προσπέλαση - Αναφορά

- ❑ Τα στοιχεία ενός πίνακα αποθηκεύονται σε γειτονικές (διαδοχικές) θέσεις μνήμης.
- ❑ Ο πίνακας είναι μία δομή τυχαίας προσπέλασης
- ❑ Χαρακτηριστικά: μέγεθος πίνακα, τύπος στοιχείων



Μεταβλητές με ενδείκτη - Ανάθεση

- ❑ Οι μεταβλητές με δείκτη χρησιμοποιούνται όπως οι κοινές μεταβλητές.
 - `x=grades[3]*2;`
 - `y=(grades[i]+10)/5;`

- ❑ Οι μεταβλητές με δείκτη χρησιμοποιούνται για ανάθεση/επεξεργασία των τιμών στα στοιχεία του πίνακα.
 - `grades[3] = 14;`
 - `grades[j] = grades[j+1]+10;`

Παράδειγμα

- Θεωρούμε μονοδιάστατο πίνακα 20 ακεραίων στοιχείων (int num[20]). Να γραφεί κώδικας C που να αυξάνει τα αρνητικά στοιχεία κατά 1 και να μειώνει τα θετικά κατά 2.

```
for (i=0; i<20; i++)  
    if (num[i] < 0)  
        num[i]++;  
    else  
        num[i] -= 2;
```

Ιδέες για εξάσκηση

- ▣ Παράδειγμα αρχικοποίησης πίνακα
- ▣ Παράδειγμα εύρεσης μέσου όρου στοιχείων πίνακα
- ▣ Παράδειγμα εύρεσης μέγιστης τιμής πίνακα
- ▣ Παράδειγμα εύρεσης ελάχιστης τιμής πίνακα
- ▣ Παράδειγμα ψαξίματος σε πίνακα

Πίνακας ως όρισμα συνάρτησης

- **Δήλωση συνάρτησης** που τυπώνει τα στοιχεία πίνακα ακεραίων.

```
void displayArray(int ar[], int size);
```

- **Ορισμός** της συνάρτησης

```
void displayArray(int ar[], int size){  
    int i;  
    for(i=0;i<size;i++)  
        printf("[%d]:%d\n",i,ar[i]);  
}
```

Η συνάρτηση δέχεται ως 1^ο όρισμα ένα πίνακα ακεραίων και ως 2^ο τον αριθμό των στοιχείων του πίνακα. **Ο πίνακας περνάει by reference.** Στην πράξη περνάει ως όρισμα ένας δείκτης int στο 1^ο στοιχείο του πίνακα.

- **Κλήση** της συνάρτησης

```
int ar1[3];  
int ar2[5]={21,22,23,24,25};  
int ar3[8]= {31,32,33,34};  
  
displayArray(ar1,3);  
displayArray(ar2,5);  
displayArray(ar3,8);
```

Αλφαριθμητικά - Πίνακες

- Αλφαριθμητικό
 - ακολουθία αλφαβητικών και αριθμητικών χαρακτήρων (π.χ. Όνομα χρήστη, τίτλος βιβλίου, ISBN βιβλίου κλπ)
- Pascal -> ειδικός τύπος (string)
- C -> πίνακας χαρακτήρων με τελευταίο στοιχείο το μηδενικό χαρακτήρα ('\0')

Αλφαριθμητικά

- Εκτύπωση

```
printf("Hello");
```

```
printf("Ο αριθμός είναι %s\n", str);
```

- Εισαγωγή

```
scanf("%s", str); ή scanf("%s", &str[0]);
```

διαβάζει μέχρι το πρώτο κενό

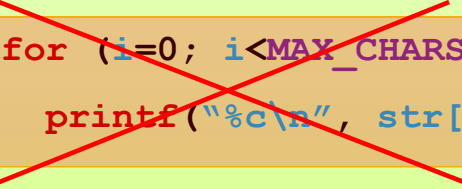
προσοχή στην υπέρβαση του μεγέθους

Παράδειγμα

```
#include <stdio.h>

#define MAX_CHARS 80


int main()
{
    char str[MAX_CHARS];
    int i;
    printf("Δώσε αριθμητικό:");
    scanf("%s", str);
    for (i=0; i<MAX_CHARS; i++)
        printf("%c\n", str[i]);
}
```



```
#include <stdio.h>

#define MAX_CHARS 80

int main()
{
    char str[MAX_CHARS];
    int i;
    printf("Δώσε αριθμητικό:");
    scanf("%s", str);
    i = 0;
    while (str[i] != '\0')
        printf("%c\n", str[i++]);
}
```



Συναρτήσεις Αλφαριθμητικών

□ <stdio.h>

```
char *gets(char *s)    // διαβάζει string από το stdin στο s
                        // και επιστρέφει το s

int puts(const char *s) // γράφει το string s στο stdout

Char * fgets(char*s, int n, FILE *fstream); // διαβάζει το πολύ n-1
//χαρακτήρες σταματώντας αν συναντήσει χαρακτήρα νέας γραμμής
```

□ <string.h>

```
int  strlen(cs)
int  strcmp(cs,ct)           int  strcmp(cs,ct, count);
char *strcpy(dest,source)    char *strcpy(dest,source, count)
char *strcat(destination, source)
```

Διάβασμα αλφαριθμητικών με κενά με χρήση scanf()

```
#include <stdio.h>
```

```
int main()
```

```
{
```

```
    char x[20];
```

```
    printf("Give: ");
```

```
    scanf("%19[^\n]s", x);
```

```
    puts(x);
```

```
}
```

<https://learn.microsoft.com/en-us/cpp/c-runtime-library/scanf-width-specification?view=msvc-170>

Ιδέες για εξάσκηση

- ▣ Παράδειγμα ανάγνωσης και εκτύπωσης συμβολοσειράς
- ▣ Παράδειγμα αρχικοποίησης συμβολοσειράς
- ▣ Παράδειγμα αντίστροφης εκτύπωσης συμβολοσειράς
- ▣ Παράδειγμα χρήσης συναρτήσεων
- ▣ Παράδειγμα που από μία συμβολοσειρά αποκόπτονται τα κενά

Δισδιάστατος Πίνακας (1)

- Αναπαράσταση 10 φοιτητών και 4 μαθημάτων: χρήση δισδιάστατου πίνακα.
 - `int grades[10][4]`
 - πίνακας 10 στοιχείων, κάθε στοιχείο πίνακας 4 ακεραίων
 - ή πίνακας 10 γραμμών και 4 στηλών.
- `grades[0][1]` -> βαθμός 2ου μαθήματος, πρώτου φοιτητή
- Αποθήκευση στη μνήμη κατά γραμμές (σαν μονοδιάστατος)

Διοδιάστατος Πίνακας (2)

Μαθήματα (στήλες)

Φοιτητές (γραμμές)

	0	1	2
0	14	15	18
1	12	14	16
2	18	15	17
3	13	19	18

```
// Αρχικοποίηση  
int grades[4][3]=  
{  
    {14,15,18},  
    {12,14,16},  
    {18,15,17},  
    {13,19,18}  
}
```

Ιδέες για εξάσκηση

- ▣ Πολλαπλασιασμός πινάκων
- ▣ Υλοποίηση λεξικού μετάφρασης όρων