

Προγραμματισμός Η/Υ Ι

06. Πίνακες (Μέρος Α)

ΕΑΡΙΝΟ ΕΞΑΜΗΝΟ 2019-2020 | ΤΜΗΜΑ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ Τ.Ε.

Συλλογή δεδομένων

- ❖ για να κάνουμε κάτι ενδιαφέρον σε ένα πρόγραμμα, χρειαζόμαστε μία **συλλογή δεδομένων** στα οποία θα δουλέψουμε, π.χ.
 - ▶ κατάλογος τηλεφώνων
 - ▶ λίστα με τα μέλη μιας ποδοσφαιρικής ομάδας
 - ▶ λίστα βιβλίων που διαβάσαμε την τελευταία χρονιά
 - ▶ έναν πίνακα μετεωρολογικών προβλέψεων
- ❖ στα πλαίσια του παρόντος μαθήματος, θα συζητήσουμε για τους **πίνακες (arrays)** έναν από τους πιο **απλούς** και πιο **χρήσιμους** τρόπους συλλογής δεδομένων
- ✍ ο πίνακας είναι μία **δομή δεδομένων**

Συλλογή Δεδομένων

Πίνακας (Array)

- ❖ μια διατεταγμένη ακολουθία δεδομένων ίδιου τύπου, στα οποία αναφερόμαστε με ένα όνομα

- ▶ δηλαδή με μια μεταβλητή

π.χ.

- ο διπλανός πίνακας ακεραίων **A** περιέχει τα στοιχεία: {1, 4, 2, 3, 5}

- ✍ είναι ένας μονοδιάστατος πίνακας δηλαδή ένα διάνυσμα

- ο διπλανός δισδιάστατος πίνακας ακεραίων **B** περιέχει τα στοιχεία: {{1,3,4}, {5,8,2}}

A:

1	4	2	3	5
---	---	---	---	---

B:

1	3	4
5	8	2

- ❖ η ιδέα αυτής της μαζικής αναφοράς στοιχείων, προέρχεται από τα μαθηματικά
 - ▶ διανύσματα, πίνακες

Συλλογή Δεδομένων

Πίνακας (Array) - Υλοποίηση

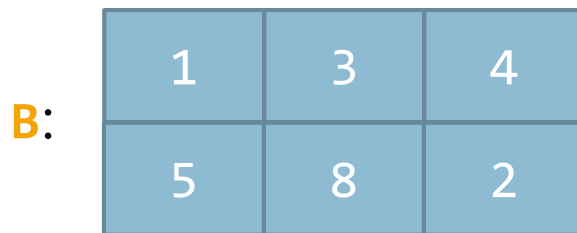
κάθε κελί της μνήμης είναι 4 byte! Γιατί;

❖ υλοποιείται με τη δέσμευση **συνεχόμενων** κελιών της κύριας μνήμης

▶ κατά τη διάρκεια της εκτέλεσης του προγράμματος

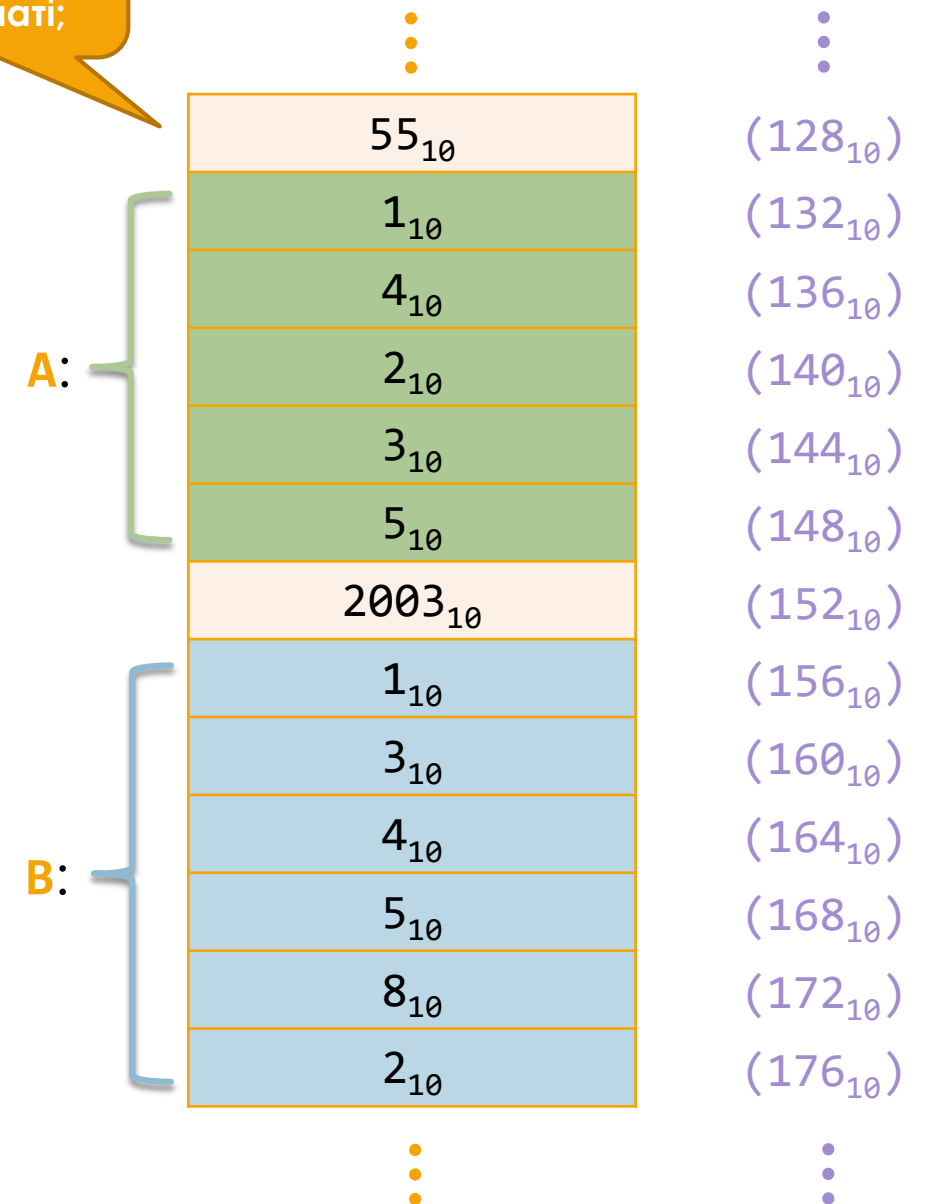
π.χ.

▶ η αναπαράσταση στη μνήμη των παρακάτω **πίνακων ακεραίων (A και B)**, παρουσιάζεται στο διπλανό σχήμα



δεδομένα / εντολές

διευθύνσεις



Συλλογή Δεδομένων

Πίνακας (Array) - Μονοδιάστατος

- ❖ ένας πίνακας μιας διάστασης είναι ένα **διάνυσμα**

π.χ.

- ▶ ένας μονοδιάστατος πίνακας με όνομα **A** που φιλοξενεί 10 πραγματικούς αριθμούς (βαθμούς φοιτητών) έχει την εξής μορφή:

A:

6.1	9.4	7.2	3.3	5.0	1.9	10.0	5.2	6.7	8.0
-----	-----	-----	-----	-----	-----	------	-----	-----	-----

βαθμός 1^{ου}
φοιτητή

βαθμός 2^{ου}
φοιτητή

βαθμός 10^{ου}
φοιτητή

Συλλογή Δεδομένων

Πίνακας (Array) - Μονοδιάστατος - Δήλωση

- ❖ η δήλωση γίνεται ως εξής:

```
float grades[10]; // για αποθήκευση 10 πραγματικών αριθμών  
int students[20]; // για αποθήκευση 20 ακεραίων αριθμών
```

- ❖ γενικά:

```
τύπος_δεδομένων όνομα_πίνακα [πλήθος_στοιχείων]
```

- ❖ προτιμούμε να δηλώσουμε το πλήθος_στοιχείων ως σταθερά!

```
#define N 10  
float grades[N];
```

Συλλογή Δεδομένων

Πίνακας (Array) - Μονοδιάστατος - Αναφορά σε στοιχεία

- ❖ για να αναφερθούμε στο στοιχείο ενός πίνακα χρησιμοποιούμε:
 - ▶ το όνομα του πίνακα και
 - ▶ το δείκτη θέσης του στοιχείου μέσα σε αγκύλες []

	A[0]	A[1]	A[2]	A[3]	A[4]	A[5]	A[6]	A[7]	A[8]	A[9]
A:	6.1	9.4	7.2	3.3	5.0	1.9	10.0	5.2	6.7	8.0

βαθμός 1^{ου}
φοιτητή

βαθμός 2^{ου}
φοιτητή

βαθμός 10^{ου}
φοιτητή

Συλλογή Δεδομένων

Πίνακας (Array) - Μονοδιάστατος - Απόδοση τιμών

- ❖ κατά την διάρκεια της εκτέλεσης του προγράμματος

```
A[j] = 8;
```

- ❖ με τη δήλωση του πίνακα (απόδοση αρχικών τιμών)

```
int A[5] = {5,8,2,6,3}; // A[0]=5; A[1]=8; A[2]=2; A[3]=6; A[4]=3;
```

- αν δεν υπάρχουν όλες οι τιμές

```
int A[5] = {5,8,2}; // A[0]=5; A[1]=8; A[2]=2; A[3]=0; A[4]=0;
```

- απόδοση αρχικών μηδενικών τιμών

```
int A[5] = {0}; // A[0]=0; A[1]=0; A[2]=0; A[3]=0; A[4]=0;
```

- μπορεί να μη δηλωθεί το πλήθος των στοιχείων όταν αποδίδονται αρχικές τιμές:

```
int A[] = {5,8,2,6,3}; // A[0]=5; A[1]=8; A[2]=2; A[3]=6; A[4]=3;
```

Συλλογή Δεδομένων

Πίνακας (Array) - Σύνοψη

- ❖ βασικά χαρακτηριστικά πινάκων:
 1. ονομασία
 2. διάσταση (1, 2, ...πολλών διαστάσεων)
 3. τύπος δεδομένων που περιέχει (π.χ. `int`, `float`, `char`, ...)
 4. πλήθος στοιχείων σε κάθε διάσταση του
- ❖ κάθε στοιχείο του πίνακα βρίσκεται σε μία **θέση** με συγκεκριμένες **συντεταγμένες**
 - ▶ η **διάσταση** προσδιορίζει τον αριθμό των ακεραίων **δεικτών θέσης** που χρειάζονται για τις **συντεταγμένες** του στοιχείου, στον πίνακα
 - ▶ για πίνακες 1 διάστασης απαιτείται 1 δείκτης θέσης: **A[i]**
 - ▶ για πίνακες 2 διαστάσεων απαιτούνται 2 δείκτες θέσης: **A[i][j]**
 - ▶ κ.ο.κ

Συλλογή Δεδομένων

Πίνακας (Array) - Σύνοψη II

- ❖ μια αρκετά **διαδεδομένη** δομή δεδομένων
 - ▶ προσφέρεται από σχεδόν **κάθε** γλώσσα προγραμματισμού
 - ▶ επιτρέπει την **αποτελεσματική** οργάνωση δεδομένων **ίδιου** τύπου
- ❖ η **επεξεργασία** των δεδομένων του γίνεται **μαζικά** με χρήση των εντολών **επανάληψης**
✍️ όπως θα δούμε στη συνέχεια στα παραδείγματα...

Συλλογή Δεδομένων

Πίνακας (Array) - Τεχνικές λεπτομέρειες

- ❖ όταν ο πίνακας δηλώνεται, τα στοιχεία του έχουν **ακαθόριστες** τιμές
 - ▶ είναι **απαραίτητο** πάντα να αρχικοποιούμε τα στοιχεία του πίνακα!
- ❖ το **μέγιστο πλήθος στοιχείων** εξαρτάται από το **μεταγλωττιστή**
 - ▶ μπορούμε να **βρούμε** το **μέγεθος** του πίνακα με τον τελεστή **sizeof()**:

```
sizeof(grades)/sizeof(float) // πλήθος στοιχείων N
```

- ❖ ο **δείκτης θέσης** εκτός από **ακέραιος** μπορεί να είναι **μεταβλητή** ή **αριθμητική έκφραση**

▶ π.χ.

1. **j = 2;**
2. **i = 4;**
3. **grades[i] = 6.5; // το 5^ο στοιχείο**
4. **grades[i+j] = 9.3; // το 7^ο στοιχείο**

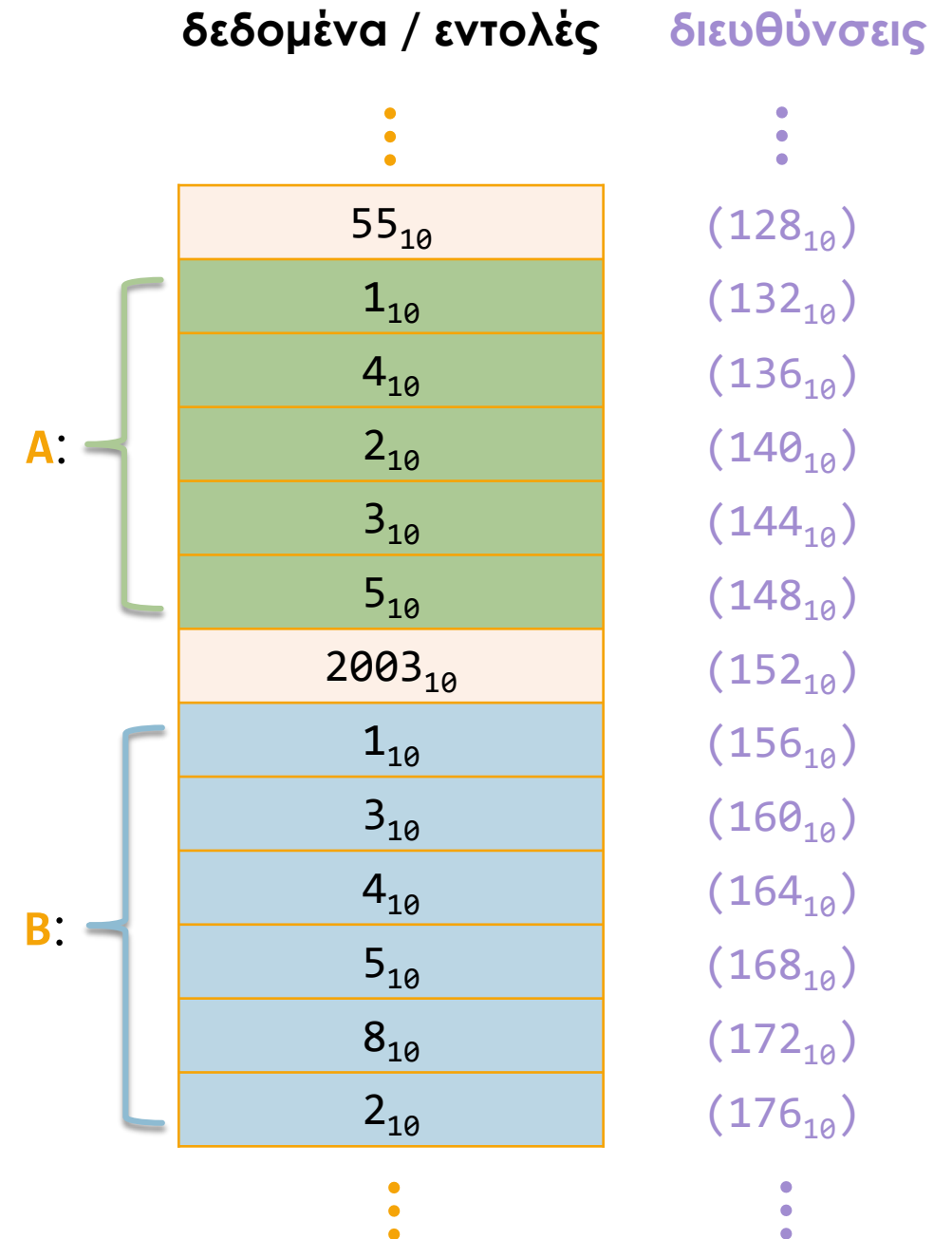
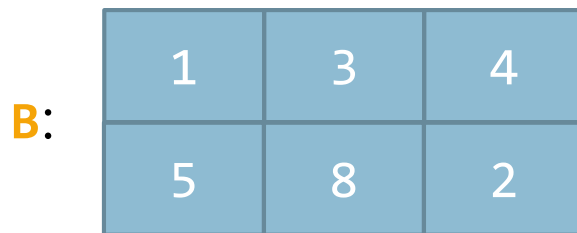
Συλλογή Δεδομένων

Πίνακας (Array) - Τεχνικές λεπτομέρειες II

- ❖ το **όνομα** του πίνακα (χωρίς τις αγκύλες) περιέχει τη **διεύθυνση** του 1^{ου} στοιχείου του πίνακα (δείκτης μνήμης)

```
printf("%d", A); // εκτυπώνει:132
```

```
printf("%d", B); // εκτυπώνει:156
```



Μονοδιάστατοι Πίνακες

Παραδείγματα

Μονοδιάστατοι Πίνακες

Παράδειγμα 1: Εκτύπωση στοιχείων πίνακα πραγματικών αριθμών

```
1. #include <stdio.h>
2. #define N 5
3. main ()
4. {
5.     int i;
6.     float A[N] = {1.2, 3.5, 2.6, 7.2, 5.1};
7.
8.     printf("Array A: \n");
9.     for (i = 0; i < N; i++) {
10.         printf("A[%d]: %4.2f\n", i, A[i]);
11.     }
12. }
```

? ποιες αλλαγές πρέπει να γίνουν αν ο **A** είναι πίνακας ακεραίων;

Μονοδιάστατοι Πίνακες

Παράδειγμα 2: Εκτύπωση στοιχείων πίνακα ακεραίων αριθμών

```
1. #include <stdio.h>
2. #define N 5
3. main ()
4. {
5.     int i;
6.     int A[N] = {1, 3, 2, 7, 5};
7.
8.     printf("Array A: \n");
9.     for (i = 0; i < N; i++) {
10.        printf("A[%d]: %d\n", i, A[i]);
11.    }
12. }
```

Μονοδιάστατοι Πίνακες

Παράδειγμα 3: Αρχικοποίηση στοιχείων

```
1. #include <stdio.h>
2. #define N 5
3. main ()
4. {
5.     int i, A[N] = {1, 3}, B[N] = {0};
6.
7.     printf("Array A: \n");
8.     for (i = 0; i < N; i++) {
9.         printf("A[%d]: %d\n", i, A[i]);
10.    }
11.
12.    printf("Array B: \n");
13.    for (i = 0; i < N; i++) {
14.        printf("B[%d]: %d\n", i, B[i]);
15.    }
16. }
```

? ΤΙ ΕΚΤΥΠΩΝΕΙ ΤΟ ΠΡΟΓΡΑΜΜΑ;

Μονοδιάστατοι Πίνακες

Παράδειγμα 4: Αρχικοποίηση στοιχείων πίνακα από το πληκτρολόγιο

```
1. #include <stdio.h>
2. #define N 10
3. main ()
4. {
5.     int i, A[N];
6.
7.     for (i = 0; i < N; i++) {
8.         printf("Give the %d element of the array: ", i+1);
9.         scanf("%d", &A[i]);
10.    }
11.
12.    printf("Your array: \n");
13.    for (i = 0; i < N; i++) {
14.        printf("A[%d]: %d\n", i, A[i]);
15.    }
16. }
```

Μονοδιάστατοι Πίνακες

Παράδειγμα 5: Το μέγεθος του πίνακα καθορίζεται από τον χρήστη

```
1. #include <stdio.h>
2. #define Nmax 100
3. main ()
4. {
5.     int i, n, A[Nmax];
6.     printf("Enter the size of the array (from 1 to 100):");
7.     scanf("%d", &n);
8.     for (i = 0; i < n; i++) {
9.         printf("Give the %d element of the array: ", i+1);
10.        scanf("%d", &A[i]);
11.    }
12.
13.    printf("Your array: \n");
14.    for (i = 0; i < n; i++) {
15.        printf("A[%d]: %d\n", i, A[i]);
16.    }
17. }
```

Η διάσταση του πίνακα δεν μπορεί να υπερβεί ένα μέγιστο **Nmax** που ορίζεται σαν σταθερά

Μονοδιάστατοι Πίνακες

Παράδειγμα 6: Πόσα στοιχεία του πίνακα είναι μικρότερα του 100;

```
1.  #include <stdio.h>
2.  #define N 10
3.  main ()
4.  {
5.      int i, A[N], M;
6.
7.      for (i = 0; i < N; i++) {
8.          printf("Give the %d element of the array: ", i+1);
9.          scanf("%d", &A[i]);
10.     }
11.     M=0;
12.     for (i = 0; i < N; i++)
13.         if (A[i] < 100) M++;
14.     if (M==0)
15.         printf(" Δεν βρήκα κανένα στοιχείο μικρότερο του 100\n");
16.     else
17.         printf("Βρήκα %d στοιχεία μικρότερα του 100\n", M );
18. }
```

Σύνοψη

- ▶ Συλλογή Δεδομένων
- ▶ Πίνακας
 - ▶ Υλοποίηση
 - ▶ Μονοδιάστατος
 - ▶ Δήλωση
 - ▶ Αναφορά σε στοιχεία
 - ▶ Απόδοση τιμών
 - ▶ Παραδείγματα