



Δ.Π.Θ Δομημένος Προγραμματισμός

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ & ΔΙΟΙΚΗΣΗΣ

Εισαγωγή στους Δείκτες

Dr. Αθανάσιος Μπαλαφούτης
Εργαστηριακό Διδακτικό Προσωπικό
Τομέας Συστημάτων Παραγωγής
Εργαστήριο Ρομποτικής και Αυτοματισμών
abalafou@pme.duth.gr
Γραφείο 304, τηλ.: 25410 – 79892

Μεταβλητές και Μνήμη RAM

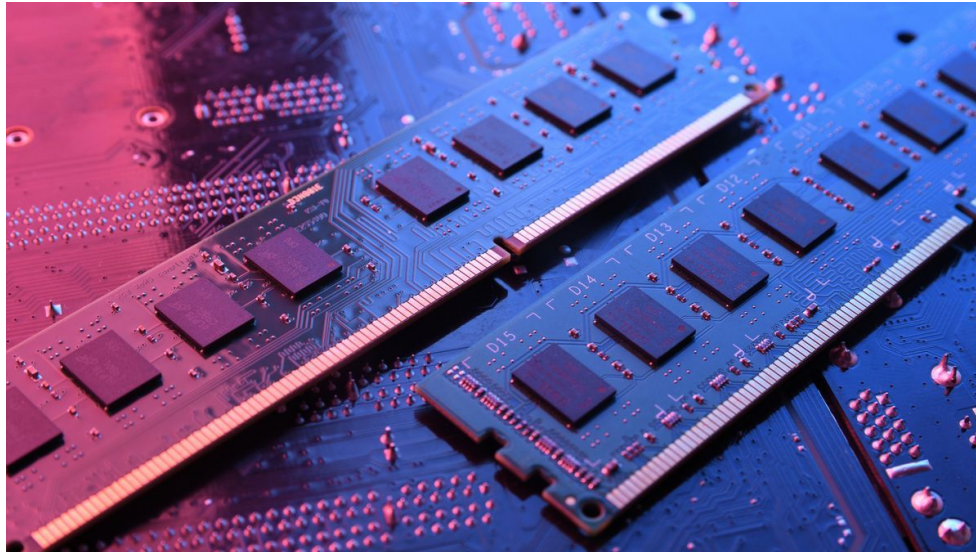


Δ.Π.Θ

Δομημένος Προγραμματισμός

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ & ΔΙΟΙΚΗΣΗΣ

Για κατανοήσουμε την έννοια των **Δεικτών**, θα πρέπει πρώτα να αναλύσουμε πώς οι μεταβλητές ενός προγράμματος αποθηκεύονται στη μνήμη RAM



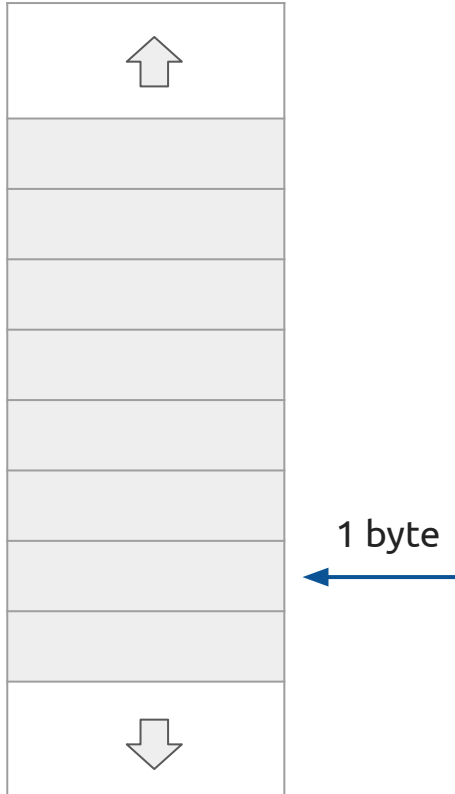
Μνήμη RAM



Δ.Π.Θ

Δομημένος Προγραμματισμός

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ & ΔΙΟΙΚΗΣΗΣ



- Η μνήμη RAM είναι χωρισμένη σε θέσεις/κελιά
- Κάθε θέση/κελί, έχει μέγεθος 1 byte (1byte = 8 bit)

Μνήμη RAM



Δ.Π.Θ

Δομημένος Προγραμματισμός

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ & ΔΙΟΙΚΗΣΗΣ

(Διεύθυνση)	
208	
207	
206	
205	
204	
203	
202	
201	
	

- Η μνήμη RAM είναι χωρισμένη σε θέσεις/κελιά
- Κάθε θέση/κελί, έχει μέγεθος 1 byte (1byte = 8 bit)
- Κάθε θέση μνήμης έχει μια διεύθυνση (ένα μοναδικό όνομα)

Μεταβλητές και Μνήμη RAM



Δ.Π.Θ

Δομημένος Προγραμματισμός

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ & ΔΙΟΙΚΗΣΗΣ

(Διεύθυνση)	
208	
207	
206	
205	
204	
203	
202	
201	
	

Εάν δηλώσω μια ακέραια μεταβλητή σε ένα πρόγραμμα:

```
#include <stdio.h>
int main(){
    int a;
}
```

Μεταβλητές και Μνήμη RAM



Δ.Π.Θ

Δομημένος Προγραμματισμός

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ & ΔΙΟΙΚΗΣΗΣ

(Διεύθυνση)	↑
208	
207	
206	
205	
204	
203	
202	
201	
	↓

Εάν δηλώσω μια ακέραια μεταβλητή σε ένα πρόγραμμα:

```
#include <stdio.h>
int main(){
    int a;
}
```

Κατά την εκτέλεση του προγράμματος, ο υπολογιστής θα δεσμεύσει μια ποσότητα μνήμης για να αποθηκεύσει τη μεταβλητή **a**.

Πόσο χώρο θα δεσμεύσει;

Μεταβλητές και Μνήμη RAM



Δ.Π.Θ

Δομημένος Προγραμματισμός

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ & ΔΙΟΙΚΗΣΗΣ

(Διεύθυνση)	↑
208	
207	
206	
205	
204	
203	
202	
201	
	↓

Εάν δηλώσω μια ακέραια μεταβλητή σε ένα πρόγραμμα:

```
#include <stdio.h>
int main(){
    int a;
}
```

Υπενθύμιση:

int: 4 bytes
char: 1 byte
float: 4 bytes
double: 8 bytes

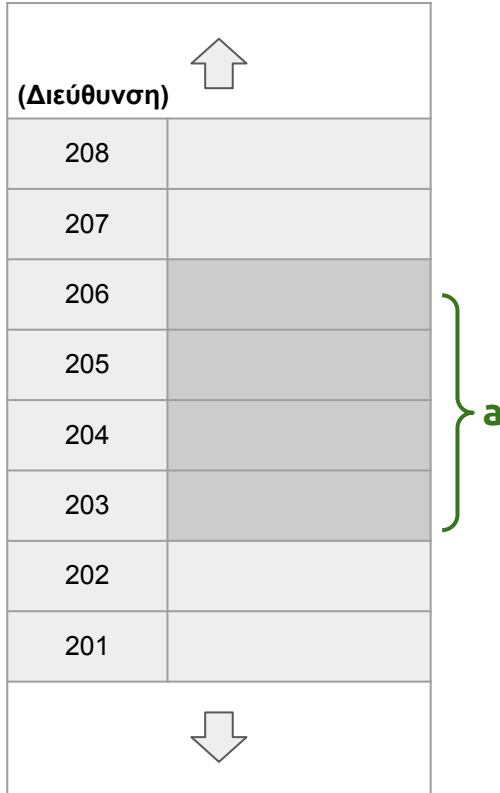
Μεταβλητές και Μνήμη RAM



Δ.Π.Θ

Δομημένος Προγραμματισμός

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ & ΔΙΟΙΚΗΣΗΣ



Εάν δηλώσω μια ακέραια μεταβλητή σε ένα πρόγραμμα:

```
#include <stdio.h>
int main(){
    int a;
}
```

Επομένως για την ακέραια μεταβλητή **a**, θα δεσμεύσει **4 bytes**

Μεταβλητές και Μνήμη RAM



Δ.Π.Θ

Δομημένος Προγραμματισμός

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ & ΔΙΟΙΚΗΣΗΣ

(Διεύθυνση)	↑		
208			
207			
206	}		
205		a	
204			}
203			
202			
201			
	↓		

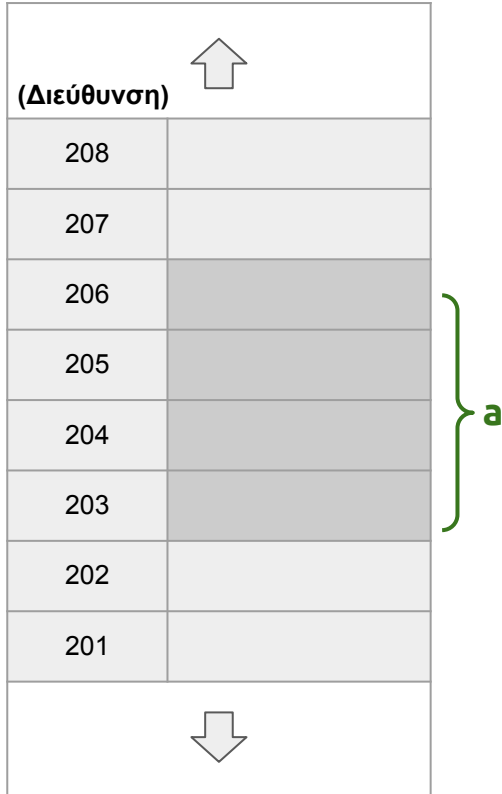
Εάν δηλώσω μια ακέραια μεταβλητή σε ένα πρόγραμμα:

```
#include <stdio.h>
int main(){
    int a;
}
```

Για να θυμάται ο υπολογιστής σε ποια θέση μνήμης αποθηκεύει τις μεταβλητές, χρησιμοποιεί ένα **lookup table**:

a int 203

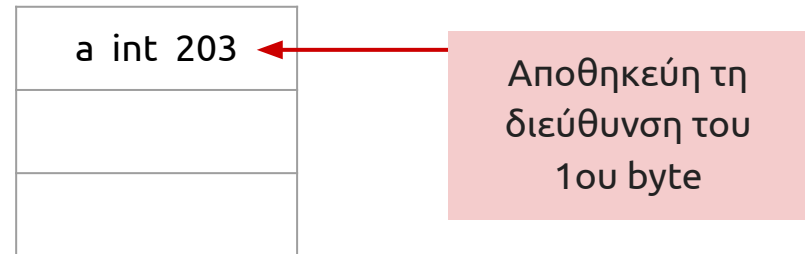
Μεταβλητές και Μνήμη RAM



Εάν δηλώσω μια ακέραια μεταβλητή σε ένα πρόγραμμα:

```
#include <stdio.h>
int main(){
    int a;
}
```

Για να θυμάται ο υπολογιστής σε ποια θέση μνήμης αποθηκεύει τις μεταβλητές, χρησιμοποιεί ένα **lookup table**:



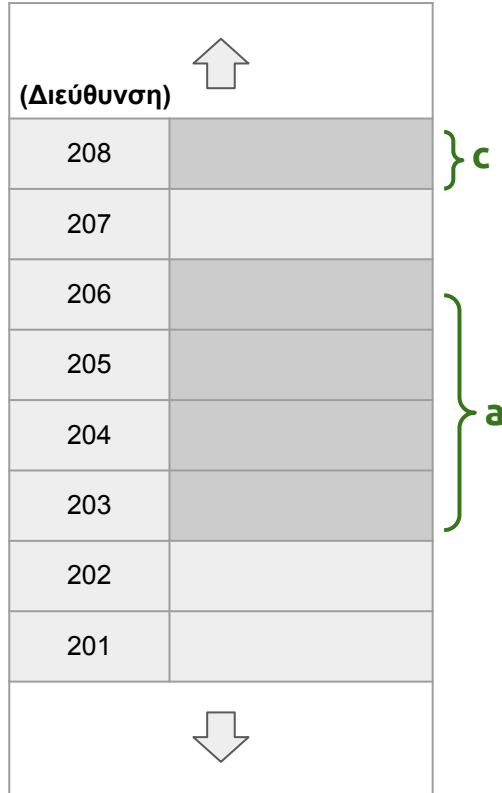
Μεταβλητές και Μνήμη RAM



Δ.Π.Θ

Δομημένος Προγραμματισμός

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ & ΔΙΟΙΚΗΣΗΣ



Εάν δηλώσω και μια μεταβλητή χαρακτήρα (char):

```
#include <stdio.h>
int main(){
    int a;
    char c;
}
```

Για να θυμάται ο υπολογιστής σε ποια θέση μνήμης αποθηκεύει τις μεταβλητές, χρησιμοποιεί ένα **lookup table**:

a int 203
c char 208

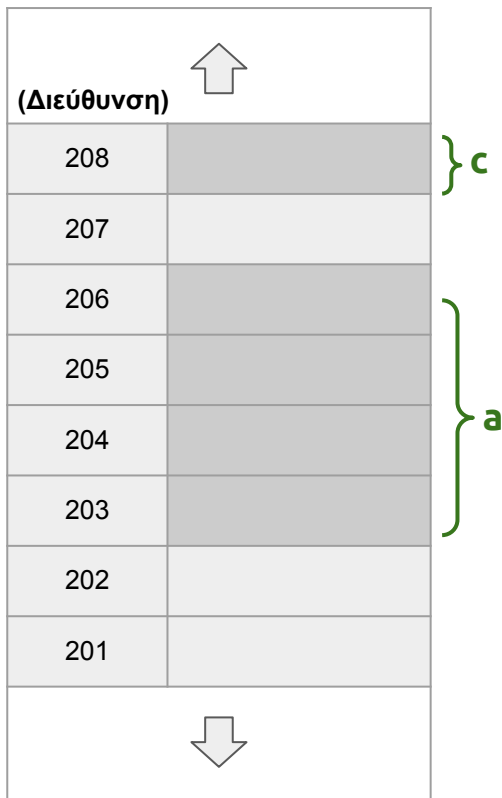
Μεταβλητές και Μνήμη RAM



Δ.Π.Θ

Δομημένος Προγραμματισμός

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ & ΔΙΟΙΚΗΣΗΣ



Αν αναθέσω στη μεταβλητή a την τιμή 5:

```
#include <stdio.h>
int main(){
    int a;
    char c;
    a = 5;
}
```

Lookup table

a	int	203
c	char	208

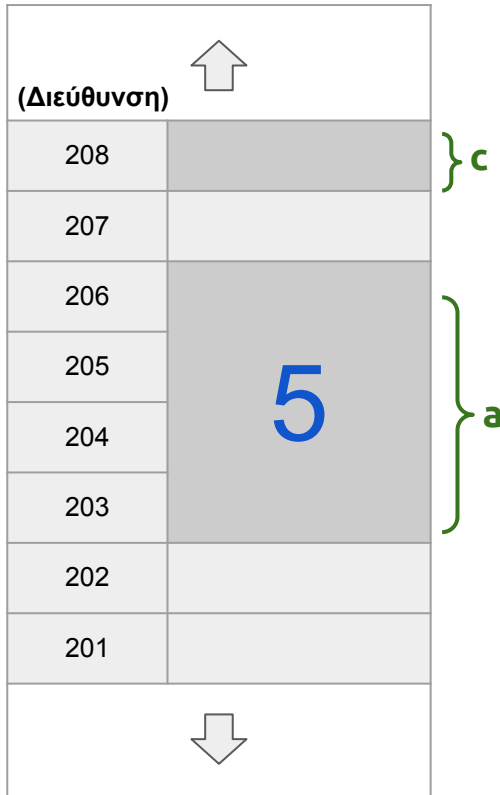
Μεταβλητές και Μνήμη RAM



Δ.Π.Θ

Δομημένος Προγραμματισμός

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ & ΔΙΟΙΚΗΣΗΣ



Αν αναθέσω στη μεταβλητή a την τιμή 5:

```
#include <stdio.h>
int main(){
    int a;
    char c;
    a = 5;
}
```

Lookup table

a	int	203
c	char	208

Το πρόγραμμα εκτελεί τις παρακάτω ενέργειες:

- Εντοπίζει στο Lookup table, τη θέση μνήμης για τη μεταβλητή a.
- Καταχωρεί σε αυτή τη θέση μνήμης της τιμή 5.

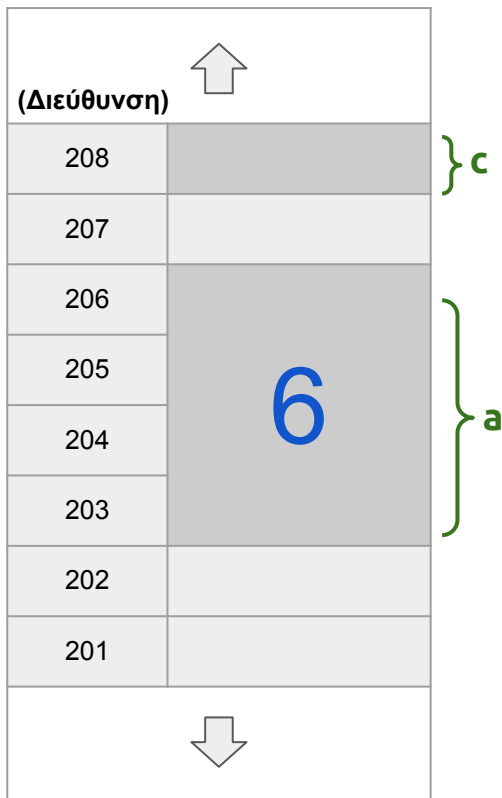
Μεταβλητές και Μνήμη RAM



Δ.Π.Θ

Δομημένος Προγραμματισμός

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ & ΔΙΟΙΚΗΣΗΣ



Αν αυξήσω κατά 1 την τιμή του a:

```
#include <stdio.h>
int main(){
    int a;
    char c;
    a = 5;
    a++;
}
```

Lookup table

a	int	203
c	char	208

Το πρόγραμμα εκτελεί τις παρακάτω ενέργειες:

- Εντοπίζει στο Lookup table, τη θέση μνήμης για τη μεταβλητή a.
- Καταχωρεί σε αυτή τη θέση μνήμης της τιμή $5+1=6$.

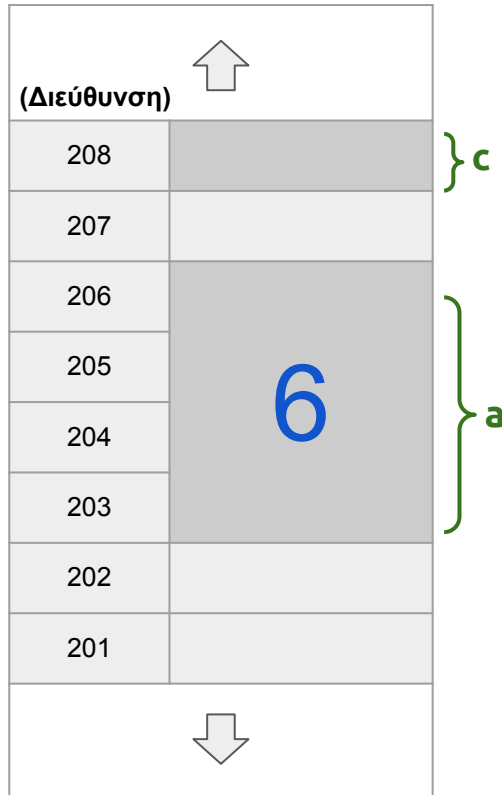
Μεταβλητές και Μνήμη RAM



Δ.Π.Θ

Δομημένος Προγραμματισμός

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ & ΔΙΟΙΚΗΣΗΣ



Αν αυξήσω κατά 1 την τιμή του a:

```
#include <stdio.h>
int main(){
    int a;
    char c;
    a = 5;
    a++;
}
```

Lookup table

a	int	203
c	char	208

Μπορούμε εμείς ως προγραμματιστές να γνωρίζουμε τη διεύθυνση μνήμης που έχει αποθηκευτεί μια μεταβλητή;

Μεταβλητές και Μνήμη RAM



Δ.Π.Θ

Δομημένος Προγραμματισμός

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ & ΔΙΟΙΚΗΣΗΣ

(Διεύθυνση)	
208	
207	
206	6
205	
204	
203	
202	
201	

↑

↓

} c

} a

Μπορούμε εμείς ως προγραμματιστές να γνωρίζουμε τη διεύθυνση μνήμης που έχει αποθηκευτεί μια μεταβλητή;

```
#include <stdio.h>
int main(){
    int a;
    char c;
    a = 5;
    a++;
    printf("Διεύθυνση μνήμης a: %p\n", &a);
}
```

Lookup table

a int 203
c char 208

Δείκτες



Δ.Π.Θ

Δομημένος Προγραμματισμός

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ & ΔΙΟΙΚΗΣΗΣ

Οι **Δείκτες** είναι **μεταβλητές** που αποθηκεύουν τη **διεύθυνση μνήμης** μιας άλλης μεταβλητής.

Δείκτες



Δ.Π.Θ

Δομημένος Προγραμματισμός

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ & ΔΙΟΙΚΗΣΗΣ

Οι **Δείκτες** είναι **μεταβλητές** που αποθηκεύουν τη **διεύθυνση** μνήμης μιας άλλης μεταβλητής.

```
#include <stdio.h>
int main(){
    int a;
}
```

Διεύθυνση	Μεταβλητή
444	a = ?
443	
442	
441	
...	
204	...
203	
202	
201	

Δείκτες



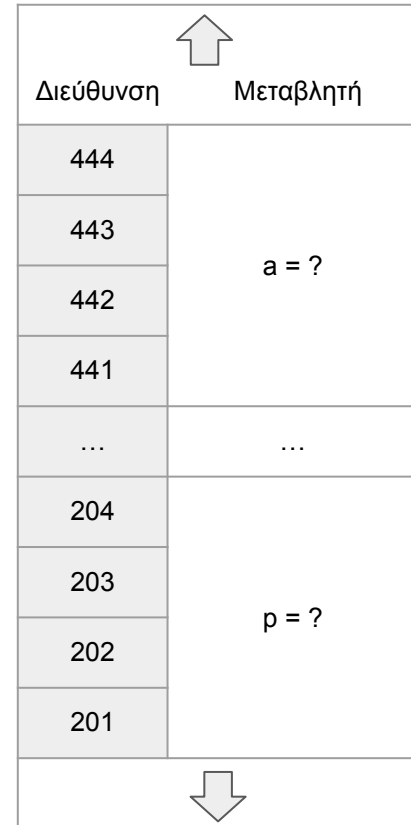
Δ.Π.Θ

Δομημένος Προγραμματισμός

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ & ΔΙΟΙΚΗΣΗΣ

Οι **Δείκτες** είναι **μεταβλητές** που αποθηκεύουν τη **διεύθυνση** μνήμης μιας άλλης μεταβλητής.

```
#include <stdio.h>
int main(){
    int a;
    int *p;
}
```



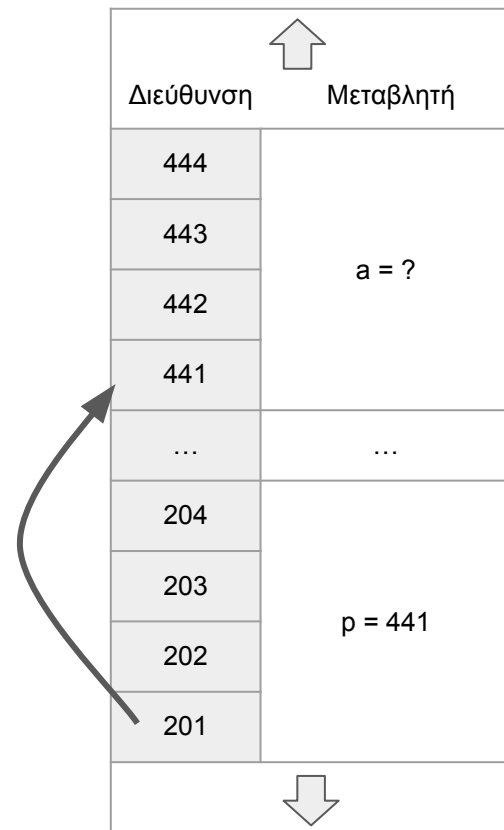
Δείκτες



Οι **Δείκτες** είναι **μεταβλητές** που αποθηκεύουν τη **διεύθυνση** μνήμης μιας άλλης μεταβλητής.

```
#include <stdio.h>
int main(){
    int a;
    int *p;

    p = &a;
}
```



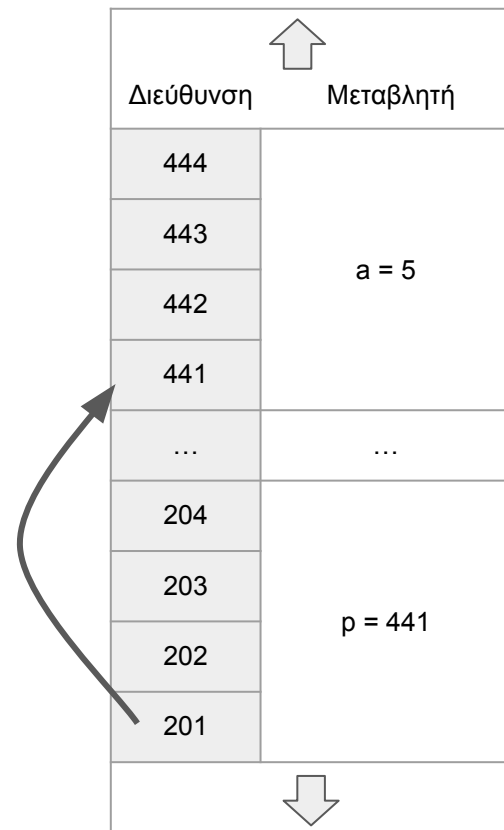
Δείκτες



Οι **Δείκτες** είναι **μεταβλητές** που αποθηκεύουν τη **διεύθυνση** μνήμης μιας άλλης μεταβλητής.

```
#include <stdio.h>
int main(){
    int a;
    int *p;

    p = &a;
    a = 5;
}
```



Δείκτες



Δ.Π.Θ

Δομημένος Προγραμματισμός

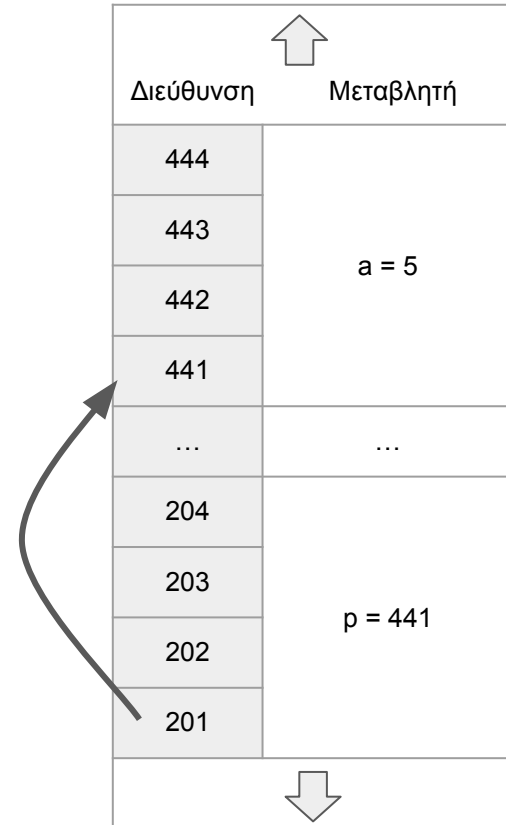
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ & ΔΙΟΙΚΗΣΗΣ

Οι **Δείκτες** είναι **μεταβλητές** που αποθηκεύουν τη **διεύθυνση** μνήμης μιας άλλης μεταβλητής.

```
#include <stdio.h>
int main(){
    int a;
    int *p;

    p = &a;
    a = 5;

    printf("Τιμή του a: %d\n", a);
}
```



Δείκτες

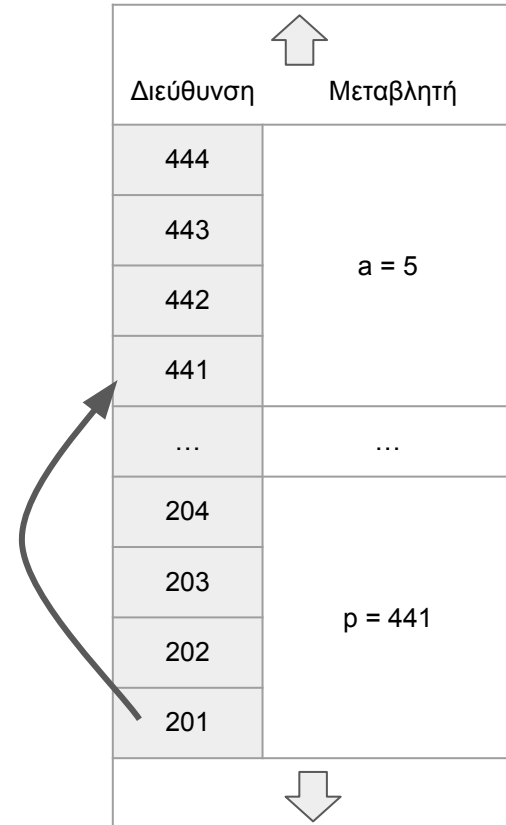


Οι **Δείκτες** είναι **μεταβλητές** που αποθηκεύουν τη **διεύθυνση** μνήμης μιας άλλης μεταβλητής.

```
#include <stdio.h>
int main(){
    int a;
    int *p;

    p = &a;
    a = 5;

    printf("Τιμή του a: %d\n", a);
}
```



Δείκτες



Δ.Π.Θ

Δομημένος Προγραμματισμός

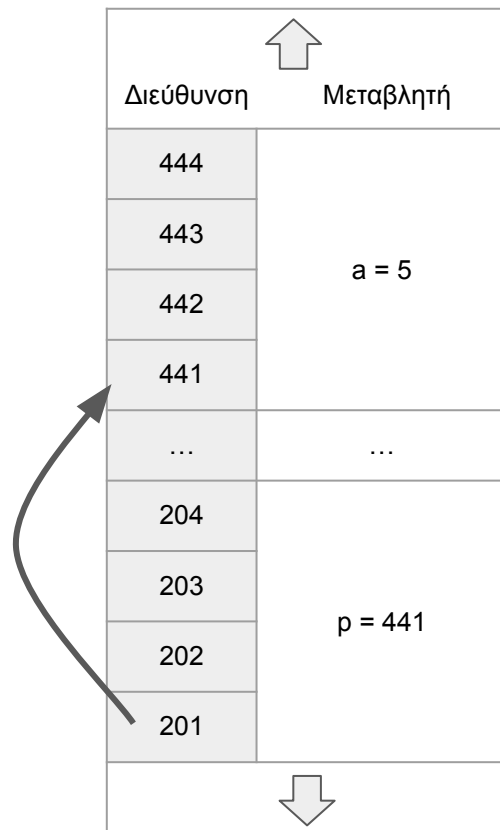
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ & ΔΙΟΙΚΗΣΗΣ

Οι **Δείκτες** είναι **μεταβλητές** που αποθηκεύουν τη **διεύθυνση** μνήμης μιας άλλης μεταβλητής.

```
#include <stdio.h>
int main(){
    int a;
    int *p;

    p = &a;
    a = 5;

    printf("Τιμή του a: %d\n", a);
    printf("Διεύθυνση του a: %p\n", &a);
}
```



Δείκτες



Δ.Π.Θ

Δομημένος Προγραμματισμός

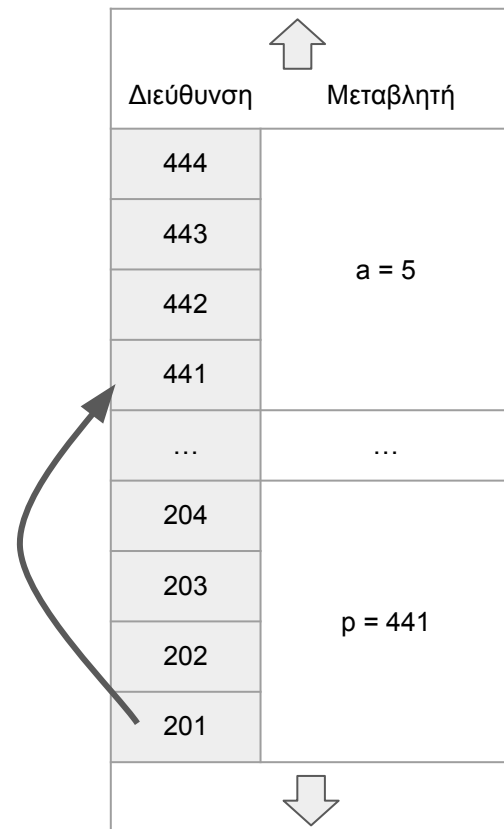
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ & ΔΙΟΙΚΗΣΗΣ

Οι **Δείκτες** είναι **μεταβλητές** που αποθηκεύουν τη **διεύθυνση** μνήμης μιας άλλης μεταβλητής.

```
#include <stdio.h>
int main(){
    int a;
    int *p;

    p = &a;
    a = 5;

    printf("Τιμή του a: %d\n", a);
    printf("Διεύθυνση του a: %p\n", &a);
}
```



Δείκτες

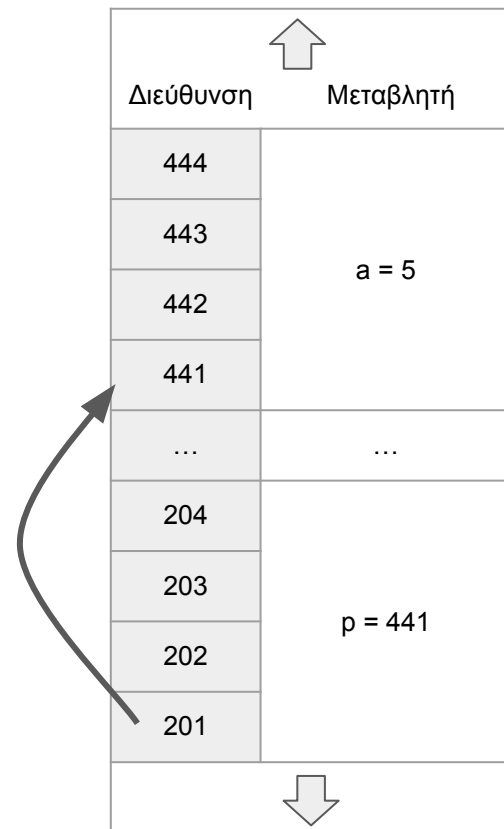


Οι **Δείκτες** είναι **μεταβλητές** που αποθηκεύουν τη **διεύθυνση** μνήμης μιας άλλης μεταβλητής.

```
#include <stdio.h>
int main(){
    int a;
    int *p;

    p = &a;
    a = 5;

    printf("Τιμή του a: %d\n", a);
    printf("Διεύθυνση του a: %p\n", &a);
    printf("Διεύθυνση του p: %p\n", &p);
}
```



Δείκτες



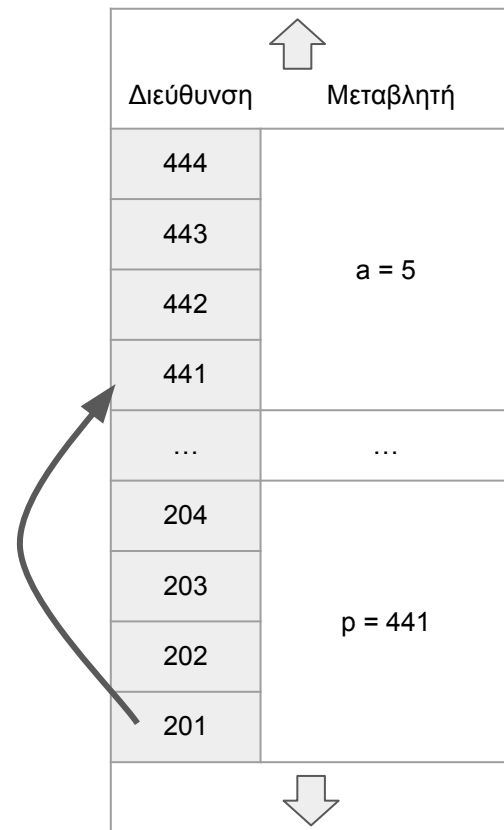
Οι **Δείκτες** είναι **μεταβλητές** που αποθηκεύουν τη **διεύθυνση** μνήμης μιας άλλης μεταβλητής.

```
#include <stdio.h>
int main(){
    int a;
    int *p;

    p = &a;
    a = 5;

    printf("Τιμή του a: %d\n", a);
    printf("Διεύθυνση του a: %p\n", &a);
    printf("Διεύθυνση του p: %p\n", &p);
}
```

201



Δείκτες

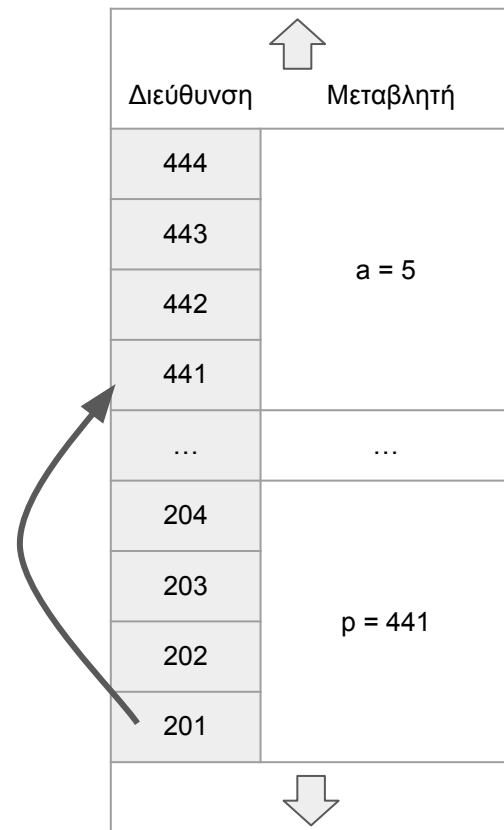


Οι **Δείκτες** είναι **μεταβλητές** που αποθηκεύουν τη **διεύθυνση** μνήμης μιας άλλης μεταβλητής.

```
#include <stdio.h>
int main(){
    int a;
    int *p;

    p = &a;
    a = 5;

    printf("Τιμή του a: %d\n", a);
    printf("Διεύθυνση του a: %p\n", &a);
    printf("Διεύθυνση του p: %p\n", &p);
    printf("Τιμή του p: %p\n", p);
}
```



Δείκτες

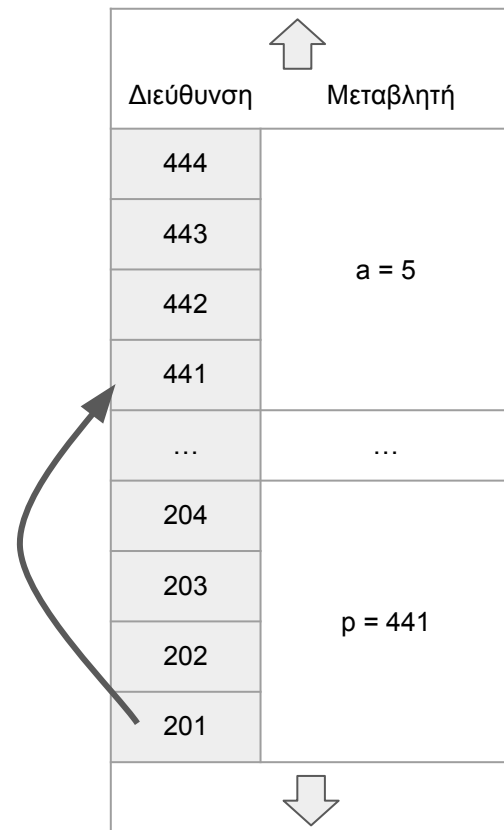


Οι **Δείκτες** είναι **μεταβλητές** που αποθηκεύουν τη **διεύθυνση** μνήμης μιας άλλης μεταβλητής.

```
#include <stdio.h>
int main(){
    int a;
    int *p;

    p = &a;
    a = 5;

    printf("Τιμή του a: %d\n", a);
    printf("Διεύθυνση του a: %p\n", &a);
    printf("Διεύθυνση του p: %p\n", &p);
    printf("Τιμή του p: %p\n", p);
}
```



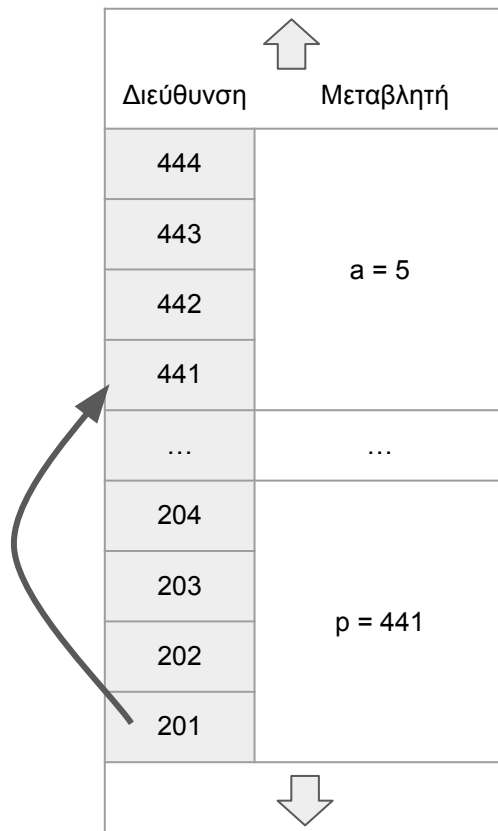
441

Δείκτες



Οι **Δείκτες** είναι **μεταβλητές** που αποθηκεύουν τη **διεύθυνση** μνήμης μιας άλλης μεταβλητής.

```
#include <stdio.h>
int main(){
    int a;
    int *p;
    p = &a;
    a = 5;
    printf("Τιμή του a: %d\n", a);
    printf("Διεύθυνση του a: %p\n", &a);
    printf("Διεύθυνση του p: %p\n", &p);
    printf("Τιμή του p: %p\n", p);
    printf("Δείκτης p: %d\n", *p);
}
```



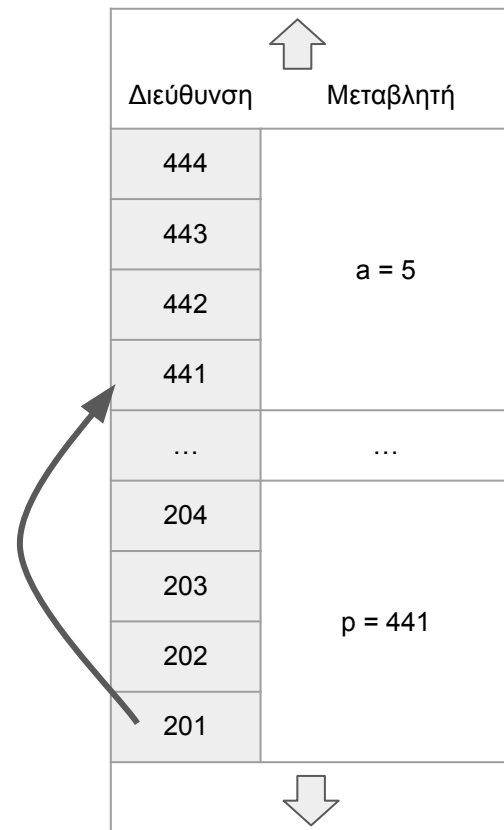
Δείκτες



Οι **Δείκτες** είναι **μεταβλητές** που αποθηκεύουν τη **διεύθυνση** μνήμης μιας άλλης μεταβλητής.

```
#include <stdio.h>
int main(){
    int a;
    int *p;
    p = &a;
    a = 5;
    printf("Τιμή του a: %d\n", a);
    printf("Διεύθυνση του a: %p\n", &a);
    printf("Διεύθυνση του p: %p\n", &p);
    printf("Τιμή του p: %p\n", p);
    printf("Δείκτης p: %d\n", *p);
}
```

5

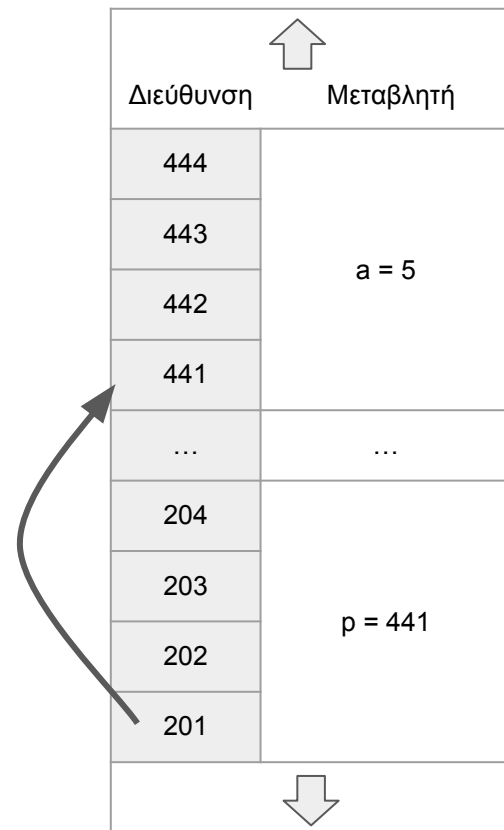


Δείκτες



Οι **Δείκτες** είναι **μεταβλητές** που αποθηκεύουν τη **διεύθυνση** μνήμης μιας άλλης μεταβλητής.

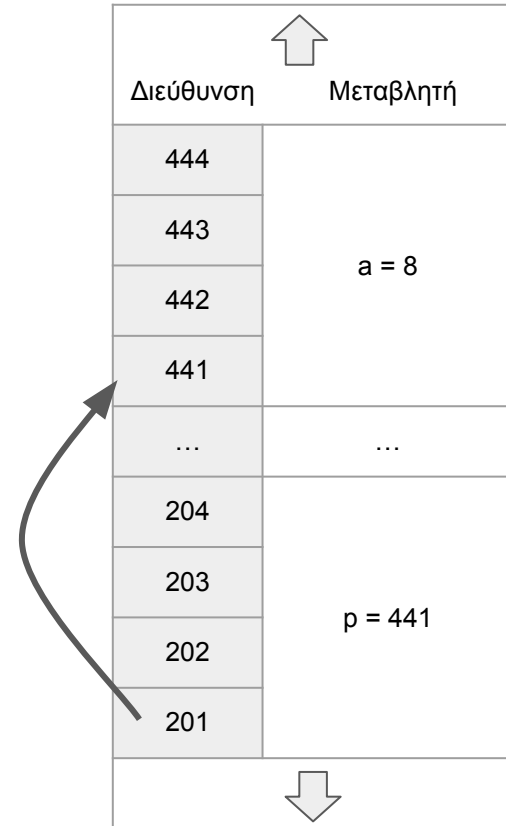
```
#include <stdio.h>
int main(){
    int a;
    int *p;
    p = &a;
    a = 5;
    printf("Τιμή του a: %d\n", a);
    printf("Διεύθυνση του a: %p\n", &a);
    printf("Διεύθυνση του p: %p\n", &p);
    printf("Τιμή του p: %p\n", p);
    printf("Δείκτης p: %d\n", *p);
    *p = 8;
    printf("a = %d\n", a);
}
```





Οι **Δείκτες** είναι **μεταβλητές** που αποθηκεύουν τη **διεύθυνση** μνήμης μιας άλλης μεταβλητής.

```
#include <stdio.h>
int main(){
    int a;
    int *p;
    p = &a;
    a = 5;
    printf("Τιμή του a: %d\n", a);
    printf("Διεύθυνση του a: %p\n", &a);
    printf("Διεύθυνση του p: %p\n", &p);
    printf("Τιμή του p: %p\n", p);
    printf("Δείκτης p: %d\n", *p);
    *p = 8;
    printf("a = %d\n", a);
}
```



8



Παραδείγματα δήλωσης δεικτών, ανάλογα με τον τύπο της μεταβλητής που δείχνουν:

```
int a;    // Ακέραια μεταβλητή
int *p;   // Δείκτης σε ακέραια μεταβλητή

char c;   // Μεταβλητή χαρακτήρα
char *p1  // Δείκτης σε μεταβλητή χαρακτήρα

double d  // Μεταβλητή μεγάλου πραγματικού αριθμού
double *p2 // Δείκτης σε μεταβλητή μεγάλου πραγματικού αριθμού
```

Δείκτες - Παράδειγμα



Δ.Π.Θ

Δομημένος Προγραμματισμός


ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ & ΔΙΟΙΚΗΣΗΣ

Τι θα εκτυπώσει το παρακάτω πρόγραμμα;

```
#include <stdio.h>
int main(){
    int a, b;
    int *p;

    a = 5;
    p = &a;

    b = 20;
    *p = b;
    printf("a = %d\n", a);
}
```



Δείκτες - Παράδειγμα



Τι θα εκτυπώσει το παρακάτω πρόγραμμα;

```
#include <stdio.h>
int main(){
    int a, b;
    int *p;

    a = 5;
    p = &a;

    b = 20;
    *p = b;
    printf("a = %d\n", a);
}
```

← 20

Ισοδύναμες εκφράσεις



Δ.Π.Θ

Δομημένος Προγραμματισμός

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ & ΔΙΟΙΚΗΣΗΣ

```
#include <stdio.h>
int main(){

    int a;
    int *p;

    a = 10;
    p = &a;

}
```



```
#include <stdio.h>
int main(){

    int a = 10;
    int *p = &a;

}
```

Αριθμητική με Δείκτες



Δ.Π.Θ

Δομημένος Προγραμματισμός

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ & ΔΙΟΙΚΗΣΗΣ

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

```
int main(){
```

```
    int a = 5;
```

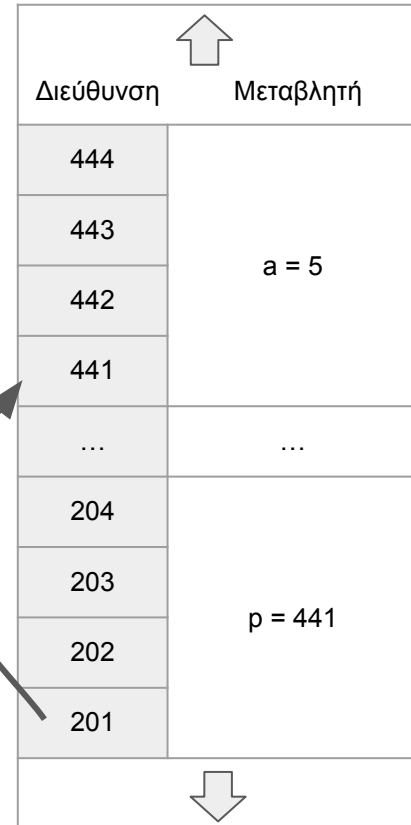
```
    int *p;
```

```
    p = &a;
```

```
    printf("Τιμή του p: %d\n", p);
```

```
    printf("Τιμή του p+1: %d\n", p+1);
```

```
}
```



Αριθμητική με Δείκτες



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

```
int main(){
```

```
    int a = 5;
```

```
    int *p;
```

```
    p = &a;
```

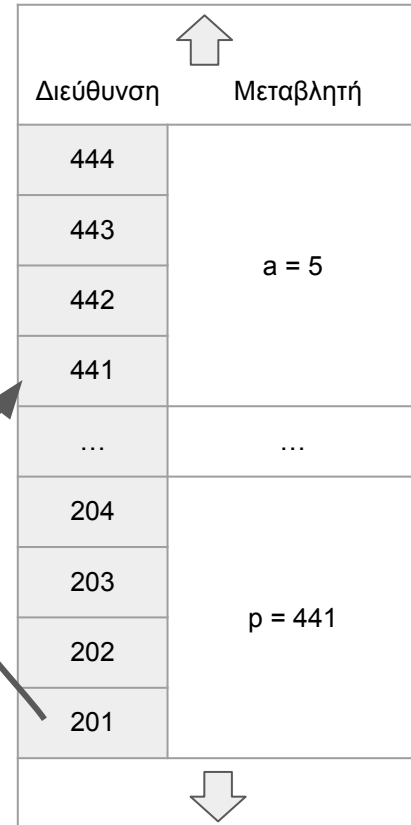
```
    printf("Τιμή του p: %d\n", p);
```

```
    printf("Τιμή του p+1: %d\n", p+1);
```

```
}
```

Ο Δείκτης p, είναι δηλωμένος ως int (4 bytes).

Η έκφραση (p+1), μετακινεί το δείκτη κατά μια θέση ακεραίου (δηλαδή 4 bytes)



Αριθμητική με Δείκτες



Δ.Π.Θ

Δομημένος Προγραμματισμός

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ & ΔΙΟΙΚΗΣΗΣ

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

```
int main(){
```

```
    int a = 5;
```

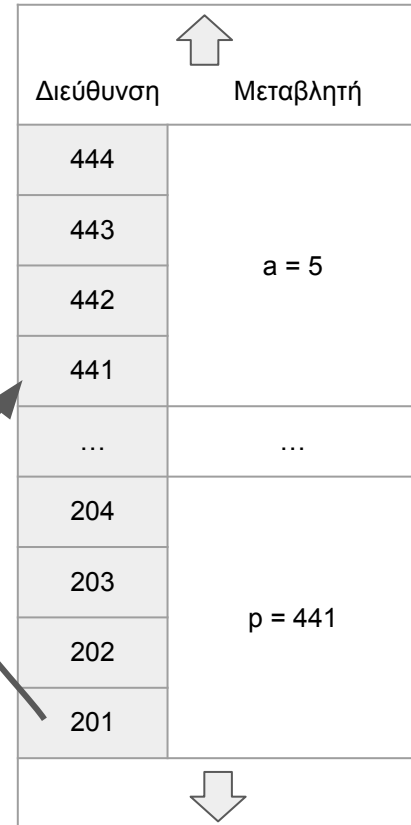
```
    int *p;
```

```
    p = &a;
```

```
    printf("Δείκτης στη θέση p: %d\n", *p);
```

```
    printf("Δείκτης στη θέση p+1: %d\n", *(p+1));
```

```
}
```



Αριθμητική με Δείκτες



Δ.Π.Θ

Δομημένος Προγραμματισμός

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ & ΔΙΟΙΚΗΣΗΣ

```
#include <stdio.h>

int main(){
    int a = 5;
    int *p;

    p = &a;

    printf("Δείκτης στη θέση p: %d\n", *p);
    printf("Δείκτης στη θέση p+1: %d\n", *(p+1));
}
```

σκουπίδια

↑ Διεύθυνση	Μεταβλητή
444	a = 5
443	
442	
441	
...	...
204	p = 441
203	
202	
201	
↓	

Ο Δείκτης **p**, δείχνει στη διεύθυνση μνήμης **441** (άρα στην τιμή **5**)

Η Δείκτης (**p+1**), δείχνει στη διεύθυνση μνήμης **445** (δεν χρησιμοποιείται από το πρόγραμμα αυτή η θέση μνήμης)

Άσκηση 2.1



Να γράψετε μια συνάρτηση `main()` που να περιέχει κατάλληλες εντολές για την επαλήθευση αυτών των παρακάτω λειτουργιών:

- A. Δήλωση και αρχικοποίηση της `next` ως μεταβλητής τύπου χαρακτήρα με τιμή `'B'`.
- B. Δήλωση και αρχικοποίηση της `current` ως μεταβλητής τύπου χαρακτήρα με τιμή `'y'`.
- C. Δήλωση του `ptr` ώστε να είναι δείκτης σε αντικείμενα τύπου `char`.
- D. Εκχώρηση της διεύθυνσης της `current` στη μεταβλητή `ptr`.
- E. Αλλαγή της τιμής του αντικειμένου στο οποίο δείχνει ο `ptr` σε `'θ'`.
- F. Εκχώρηση της διεύθυνσης της `next` στη μεταβλητή `ptr`.
- G. Αλλαγή της τιμής της μεταβλητής στην οποία δείχνει ο `ptr` σε `'d'`.
- H. Εμφάνιση της διεύθυνσης που είναι αποθηκευμένη στην `ptr`.
- I. Ποιες τιμές είναι αποθηκευμένες στις `next` και `current`.

Άσκηση 2.1 - main



Δ.Π.Θ

Δομημένος Προγραμματισμός

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ & ΔΙΟΙΚΗΣΗΣ

A. Δήλωση και αρχικοποίηση της `next` ως μεταβλητής τύπου χαρακτήρα με τιμή `'B'`?

```
#include <stdio.h>
int main() {

    return 0;
}
```

↑	Διεύθυνση	Μεταβλητή

↓		

Άσκηση 2.1 - main (Απάντηση στο Β)



Δ.Π.Θ

Δομημένος Προγραμματισμός

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ & ΔΙΟΙΚΗΣΗΣ

Β. Δήλωση και αρχικοποίηση της `current` ως μεταβλητής τύπου χαρακτήρα με τιμή `'y'`.

```
#include <stdio.h>
int main() {
    char next = 'B';
    char current = 'y';

    return 0;
}
```

↑	Διεύθυνση	Μεταβλητή
	441	next = 'B'

	333	current = 'y'

↓		

Άσκηση 2.1 - main



Δ.Π.Θ

Δομημένος Προγραμματισμός

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ & ΔΙΟΙΚΗΣΗΣ

C. Δήλωση του `ptr` ώστε να είναι δείκτης σε αντικείμενα τύπου `char`?

```
#include <stdio.h>
int main() {
    char next = 'B';
    char current = 'y';

    return 0;
}
```

↑	Διεύθυνση	Μεταβλητή
	441	next = 'B'

	333	current = 'y'

↓		

Άσκηση 2.1 - main (Απάντηση στο C)



Δ.Π.Θ

Δομημένος Προγραμματισμός

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ & ΔΙΟΙΚΗΣΗΣ

C. Δήλωση του `ptr` ώστε να είναι δείκτης σε αντικείμενα τύπου `char`

```
#include <stdio.h>
int main() {
    char next = 'B';
    char current = 'y';
    char *ptr;

    return 0;
}
```

Διεύθυνση	Μεταβλητή
441	next = 'B'
...	...
333	current = 'y'
...	...
315	ptr = (σκουπίδια)
...	...
...	...

Άσκηση 2.1 - main



Δ.Π.Θ

Δομημένος Προγραμματισμός

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ & ΔΙΟΙΚΗΣΗΣ

D. Εκχώρηση της διεύθυνσης της `current` στη μεταβλητή `ptr`?

```
#include <stdio.h>
int main() {
    char next = 'B';
    char current = 'y';
    char *ptr;

    return 0;
}
```

Διεύθυνση	Μεταβλητή
441	next = 'B'
...	...
333	current = 'y'
...	...
315	ptr = (σκουπίδια)
...	...
...	...

Άσκηση 2.1 - main (Απάντηση στο D)



Δ.Π.Θ

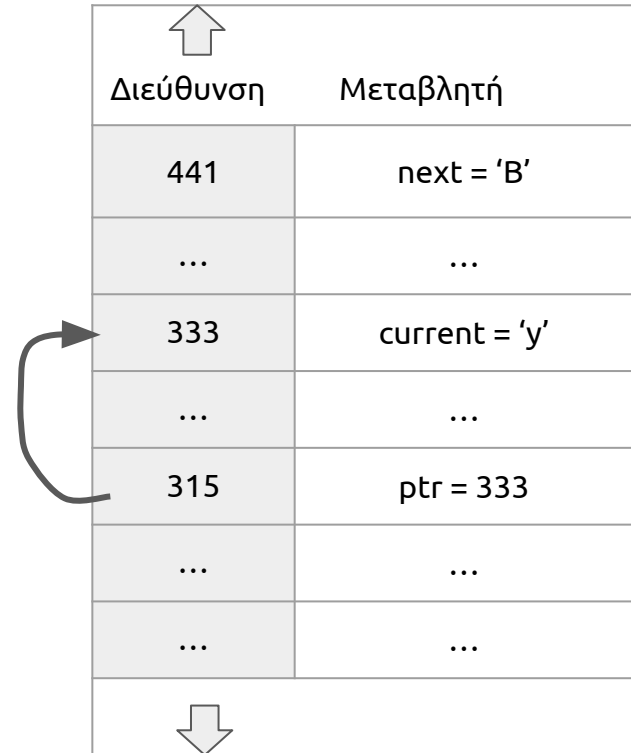
Δομημένος Προγραμματισμός

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ & ΔΙΟΙΚΗΣΗΣ

D. Εκχώρηση της διεύθυνσης της `current` στη μεταβλητή `ptr`.

```
#include <stdio.h>
int main() {
    char next = 'B';
    char current = 'y';
    char *ptr;
    ptr = &current;

    return 0;
}
```



Άσκηση 2.1 - main



Δ.Π.Θ

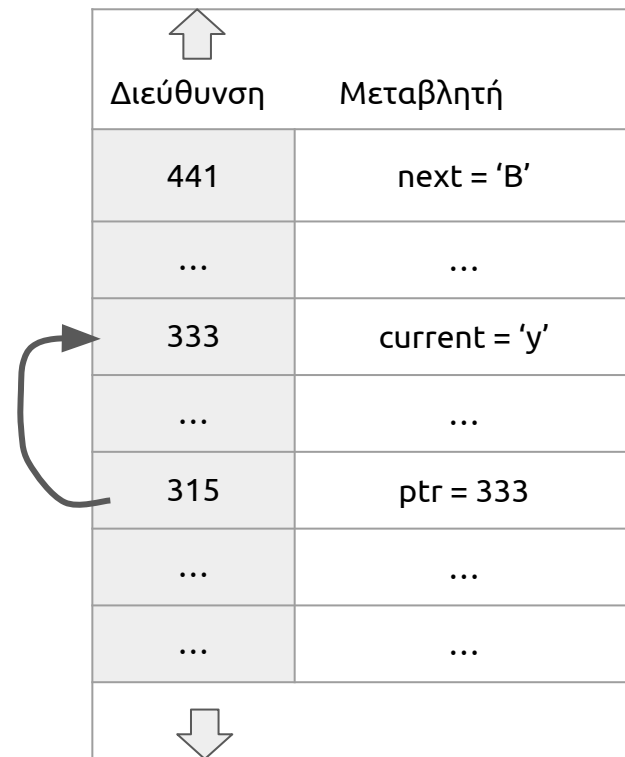
Δομημένος Προγραμματισμός

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ & ΔΙΟΙΚΗΣΗΣ

Ε. Αλλαγή της τιμής του αντικειμένου στο οποίο δείχνει ο `ptr` σε `'0'`?

```
#include <stdio.h>
int main() {
    char next = 'B';
    char current = 'y';
    char *ptr;
    ptr = &current;

    return 0;
}
```



Άσκηση 2.1 - main (Απάντηση στο Ε)



Δ.Π.Θ

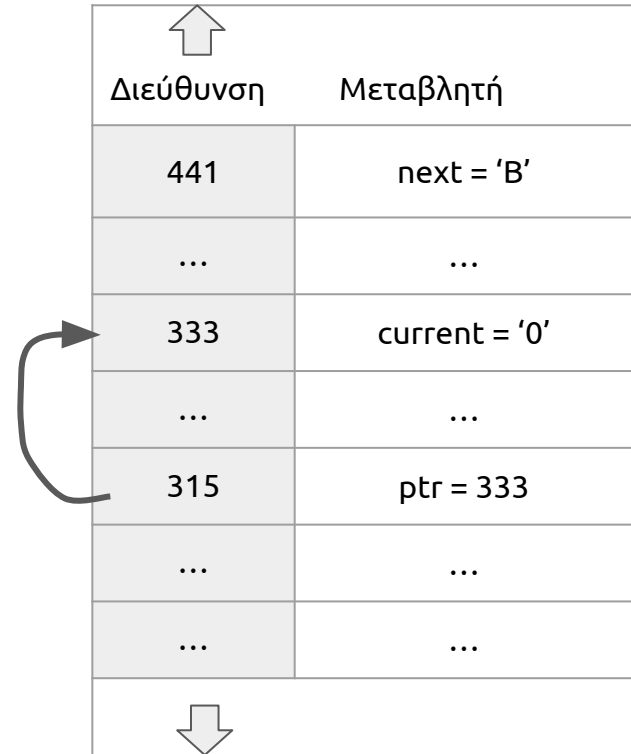
Δομημένος Προγραμματισμός

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ & ΔΙΟΙΚΗΣΗΣ

Ε. Αλλαγή της τιμής του αντικειμένου στο οποίο δείχνει ο `ptr` σε `'0'`.

```
#include <stdio.h>
int main() {
    char next = 'B';
    char current = 'y';
    char *ptr;
    ptr = &current;
    *ptr = '0';

    return 0;
}
```



Άσκηση 2.1 - main



Δ.Π.Θ

Δομημένος Προγραμματισμός

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ & ΔΙΟΙΚΗΣΗΣ

F. Εκχώρηση της διεύθυνσης της `next` στη μεταβλητή `ptr`?

```
#include <stdio.h>
int main() {
    char next = 'B';
    char current = 'y';
    char *ptr;
    ptr = &current;
    *ptr = '0';

    return 0;
}
```

Διεύθυνση	Μεταβλητή
441	next = 'B'
...	...
333	current = '0'
...	...
315	ptr = 333
...	...
...	...

Άσκηση 2.1 - main (Απάντηση στο F)



Δ.Π.Θ

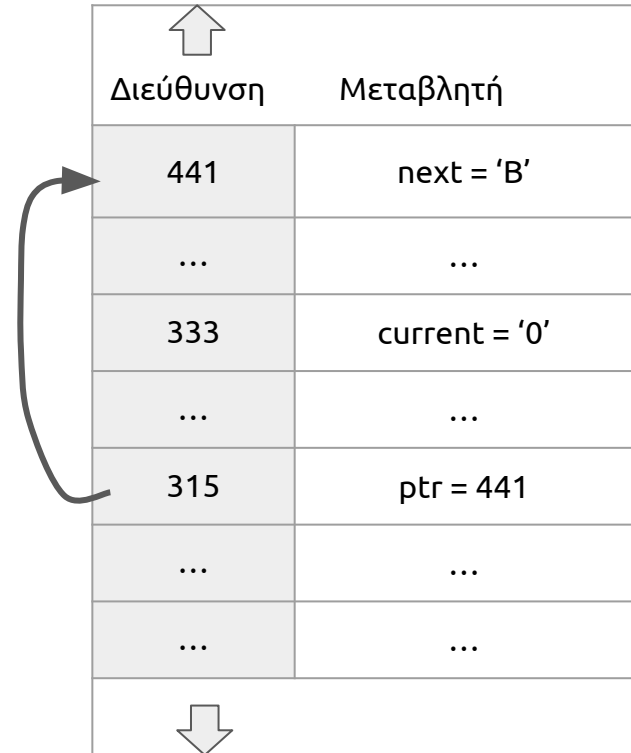
Δομημένος Προγραμματισμός

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ & ΔΙΟΙΚΗΣΗΣ

F. Εκχώρηση της διεύθυνσης της `next` στη μεταβλητή `ptr`.

```
#include <stdio.h>
int main() {
    char next = 'B';
    char current = 'y';
    char *ptr;
    ptr = &current;
    *ptr = '0';
    ptr = &next;

    return 0;
}
```



Άσκηση 2.1 - main



Δ.Π.Θ

Δομημένος Προγραμματισμός

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ & ΔΙΟΙΚΗΣΗΣ

Γ. Αλλαγή της τιμής της μεταβλητής στην οποία δείχνει ο `ptr` σε `'d'`?

```
#include <stdio.h>
int main() {
    char next = 'B';
    char current = 'y';
    char *ptr;
    ptr = &current;
    *ptr = '0';
    ptr = &next;

    return 0;
}
```

Διεύθυνση	Μεταβλητή
441	next = 'B'
...	...
333	current = '0'
...	...
315	ptr = 441
...	...
...	...

Άσκηση 2.1 - main (Απάντηση στο G)



Δ.Π.Θ

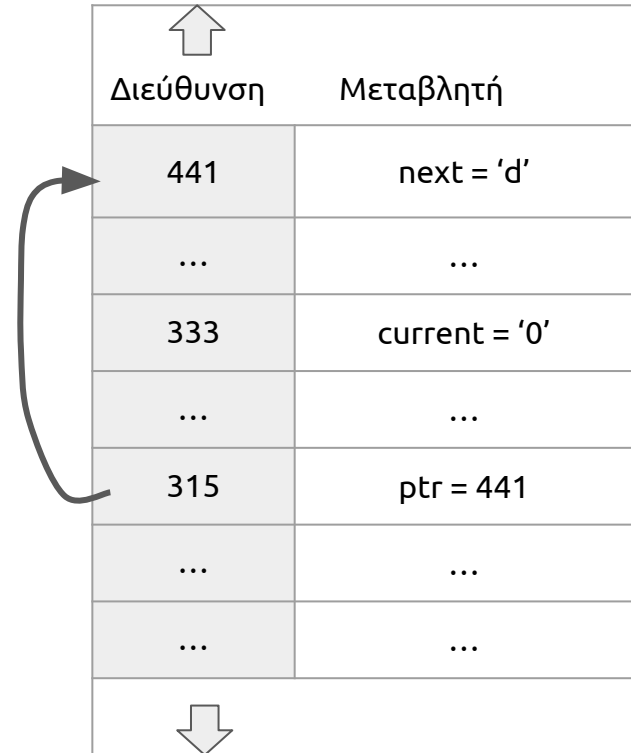
Δομημένος Προγραμματισμός

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ & ΔΙΟΙΚΗΣΗΣ

G. Αλλαγή της τιμής της μεταβλητής στην οποία δείχνει ο `ptr` σε `'d'`.

```
#include <stdio.h>
int main() {
    char next = 'B';
    char current = 'y';
    char *ptr;
    ptr = &current;
    *ptr = '0';
    ptr = &next;
    *ptr = 'd';

    return 0;
}
```



Άσκηση 2.1 - main



Δ.Π.Θ

Δομημένος Προγραμματισμός

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ & ΔΙΟΙΚΗΣΗΣ

Η. Εμφάνιση της διεύθυνσης που είναι αποθηκευμένη στην `ptr`?

```
#include <stdio.h>
int main() {
    char next = 'B';
    char current = 'y';
    char *ptr;
    ptr = &current;
    *ptr = '0';
    ptr = &next;
    *ptr = 'd';

    return 0;
}
```

Διεύθυνση	Μεταβλητή
441	next = 'd'
...	...
333	current = '0'
...	...
315	ptr = 441
...	...
...	...

Άσκηση 2.1 - main (Απάντηση στο Η)



Δ.Π.Θ

Δομημένος Προγραμματισμός

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ & ΔΙΟΙΚΗΣΗΣ

Η. Εμφάνιση της διεύθυνσης που είναι αποθηκευμένη στην `ptr`.

```
#include <stdio.h>
int main() {
    char next = 'B';
    char current = 'y';
    char *ptr;
    ptr = &current;
    *ptr = '0';
    ptr = &next;
    *ptr = 'd';
    printf("Διεύθυνση ptr: %p\n", ptr); // 441

    return 0;
}
```

Διεύθυνση	Μεταβλητή
441	next = 'd'
...	...
333	current = '0'
...	...
315	ptr = 441
...	...
...	...

Άσκηση 2.1 - main



Δ.Π.Θ

Δομημένος Προγραμματισμός

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ & ΔΙΟΙΚΗΣΗΣ

I. Ποιες τιμές είναι αποθηκευμένες στις `next` και `current`?

```
#include <stdio.h>
int main() {
    char next = 'B';
    char current = 'y';
    char *ptr;
    ptr = &current;
    *ptr = '0';
    ptr = &next;
    *ptr = 'd';
    printf("Διεύθυνση ptr: %p\n", ptr); // 441

    return 0;
}
```

Διεύθυνση	Μεταβλητή
441	next = 'd'
...	...
333	current = '0'
...	...
315	ptr = 441
...	...
...	...

Άσκηση 2.1 - main (Απάντηση στο I)



Δ.Π.Θ

Δομημένος Προγραμματισμός

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ & ΔΙΟΙΚΗΣΗΣ

I. Ποιες τιμές είναι αποθηκευμένες στις `next` και `current`?

```
#include <stdio.h>
int main() {
    char next = 'B';
    char current = 'y';
    char *ptr;
    ptr = &current;
    *ptr = '0';
    ptr = &next;
    *ptr = 'd';
    printf("Διεύθυνση ptr: %p\n", ptr); // 441
    printf("Τιμή της next: %c\n", next); // d
    printf("Τιμή της current: %c\n", current); // 0
    return 0;
}
```

Διεύθυνση	Μεταβλητή
441	next = 'd'
...	...
333	current = '0'
...	...
315	ptr = 441
...	...
...	...

Άσκηση 2.2



Δίνονται οι παρακάτω δηλώσεις:

```
int x=0;
int y=0;
int *myPtr=NULL;
int *otherPtr=NULL;
```

Με τη βοήθεια κατάλληλων εντολών `printf`, βρείτε τις τιμές των `myPtr`, `otherPtr`, `x`, και `y`, μετά από κάθε μια από τις παρακάτω γραμμές κώδικα:

1. `myPtr = &x;`
2. `otherPtr = &y;`
3. `*myPtr = 4;`
4. `*otherPtr = *myPtr;`
5. `x = 5;`
6. `otherPtr = myPtr;`
7. `*otherPtr = 6;`

Άσκηση 2.2



Δ.Π.Θ

Δομημένος Προγραμματισμός

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ & ΔΙΟΙΚΗΣΗΣ

```
#include <stdio.h>
int main() {
    int x = 0;
    int y = 0;
    int *myPtr = NULL;
    int *otherPtr = NULL;

    return 0;
}
```

↑	Διεύθυνση	Μεταβλητή
	441...444	x = 0

	401...404	y = 0

	252...255	otherPtr = NULL

	200...203	myPtr = NULL
↓		

Άσκηση 2.2



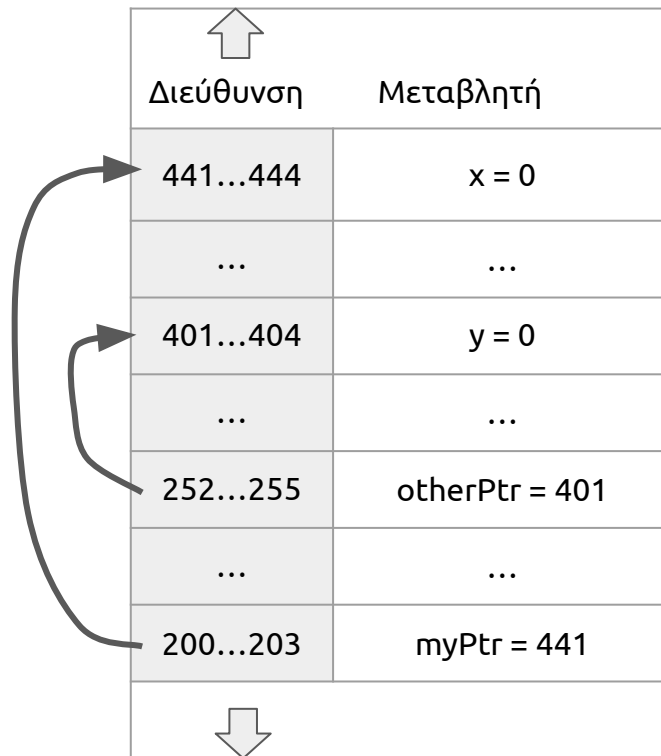
Δ.Π.Θ

Δομημένος Προγραμματισμός

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ & ΔΙΟΙΚΗΣΗΣ

```
#include <stdio.h>
int main() {
    int x = 0;
    int y = 0;
    int *myPtr = NULL;
    int *otherPtr = NULL;
    myPtr = &x;
    otherPtr = &y;

    return 0;
}
```

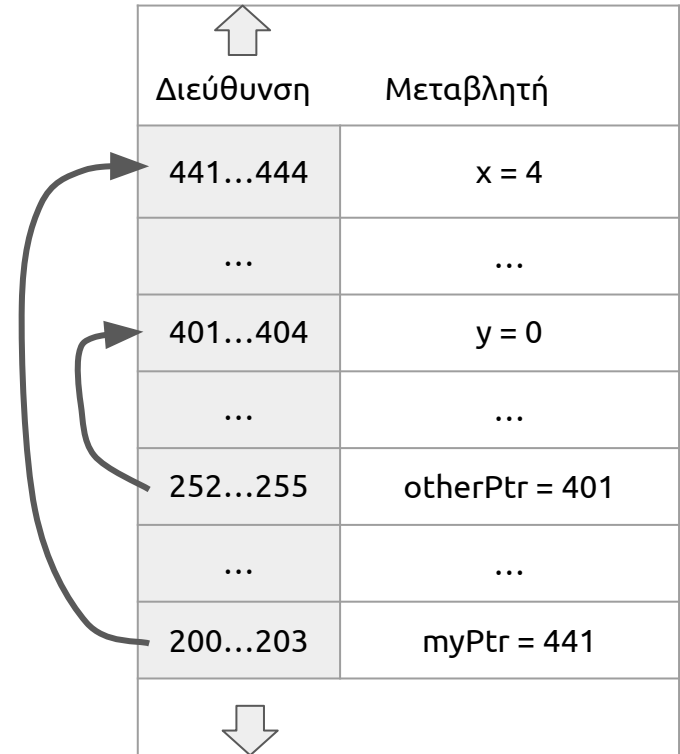


Άσκηση 2.2



```
#include <stdio.h>
int main() {
    int x = 0;
    int y = 0;
    int *myPtr = NULL;
    int *otherPtr = NULL;
    myPtr = &x;
    otherPtr = &y;
    *myPtr = 4;

    return 0;
}
```



Άσκηση 2.2



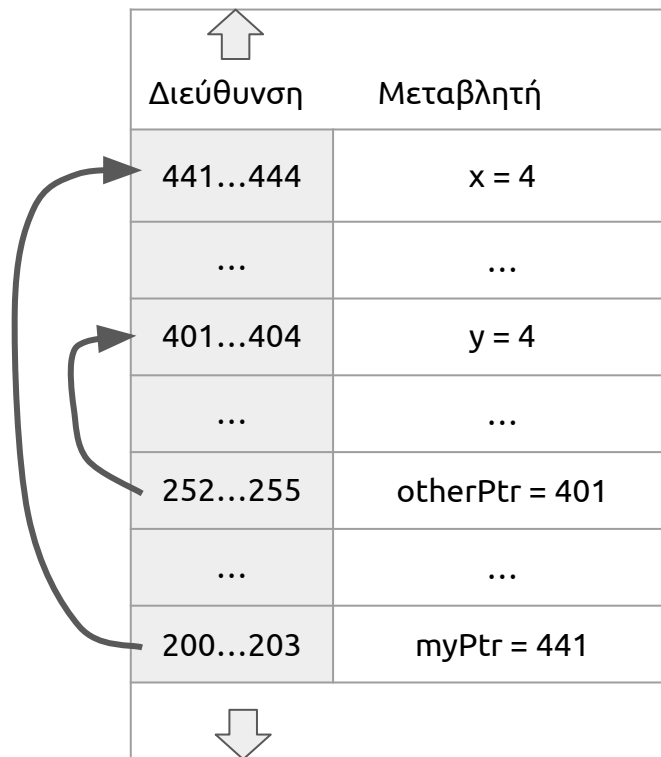
Δ.Π.Θ

Δομημένος Προγραμματισμός

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ & ΔΙΟΙΚΗΣΗΣ

```
#include <stdio.h>
int main() {
    int x = 0;
    int y = 0;
    int *myPtr = NULL;
    int *otherPtr = NULL;
    myPtr = &x;
    otherPtr = &y;
    *myPtr = 4;
    *otherPtr = *myPtr;

    return 0;
}
```



Άσκηση 2.2



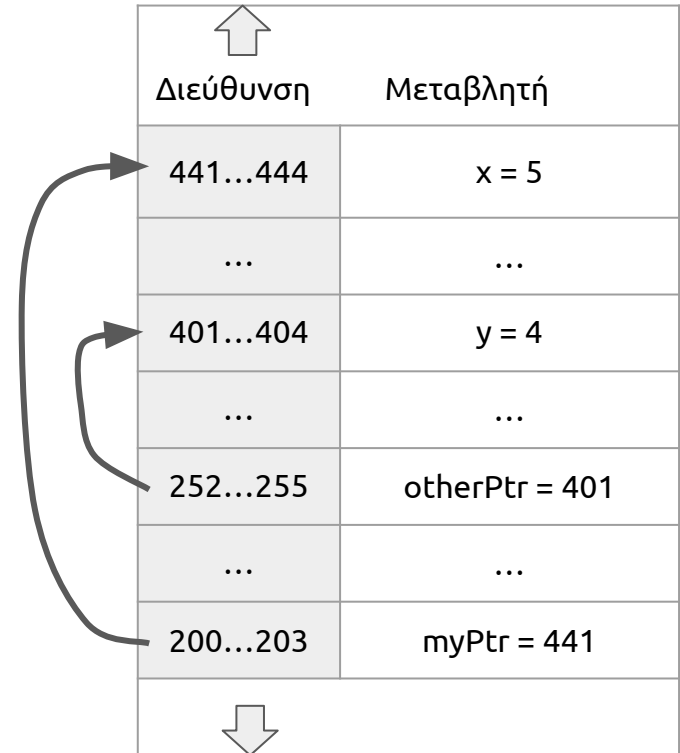
Δ.Π.Θ

Δομημένος Προγραμματισμός

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ & ΔΙΟΙΚΗΣΗΣ

```
#include <stdio.h>
int main() {
    int x = 0;
    int y = 0;
    int *myPtr = NULL;
    int *otherPtr = NULL;
    myPtr = &x;
    otherPtr = &y;
    *myPtr = 4;
    *otherPtr = *myPtr;
    x = 5;

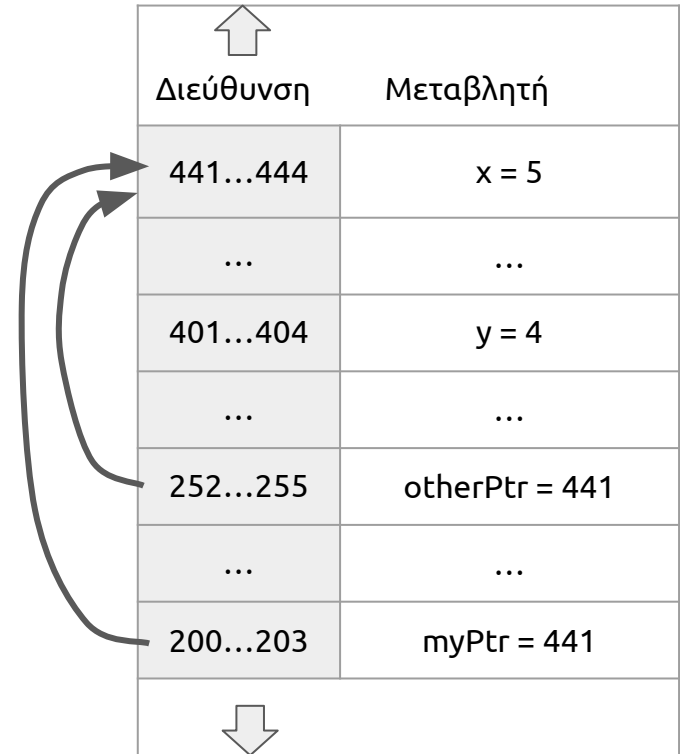
    return 0;
}
```



Άσκηση 2.2

```
#include <stdio.h>
int main() {
    int x = 0;
    int y = 0;
    int *myPtr = NULL;
    int *otherPtr = NULL;
    myPtr = &x;
    otherPtr = &y;
    *myPtr = 4;
    *otherPtr = *myPtr;
    x = 5;
    otherPtr = myPtr;

    return 0;
}
```

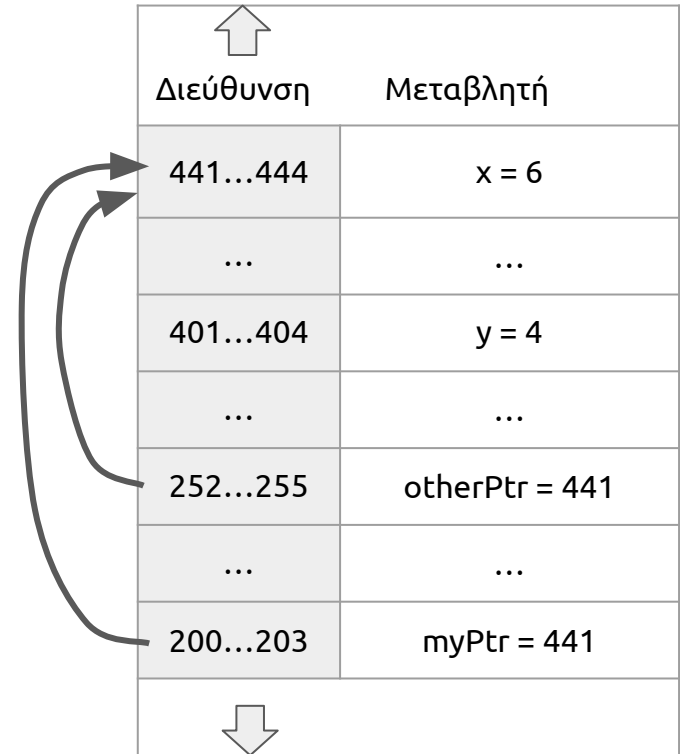


Άσκηση 2.2



```
#include <stdio.h>
int main() {
    int x = 0;
    int y = 0;
    int *myPtr = NULL;
    int *otherPtr = NULL;
    myPtr = &x;
    otherPtr = &y;
    *myPtr = 4;
    *otherPtr = *myPtr;
    x = 5;
    otherPtr = myPtr;
    *otherPtr = 6;

    return 0;
}
```



Άσκηση 2.2



```
#include <stdio.h>
int main() {
    int x = 0;
    int y = 0;
    int *myPtr = NULL;
    int *otherPtr = NULL;
    myPtr = &x;
    otherPtr = &y;
    *myPtr = 4;
    *otherPtr = *myPtr;
    x = 5;
    otherPtr = myPtr;
    *otherPtr = 6;
    printf("myPtr: %p, otherPtr: %p, x: %d, y: %d\n\n", myPtr, otherPtr, x, y);
    return 0;
}
```

Άσκηση 2.3

Τι πιστεύετε ότι θα γίνει αν εκτελέσετε το ακόλουθο πρόγραμμα και γιατί;

```
#include <stdio.h>

void main(){
    double trouble = 3.27;
    double *pt;
    *pt = 27.3;
    pt = &trouble;
}
```

Άσκηση 2.3 - segmentation fault



Δ.Π.Θ

Δομημένος Προγραμματισμός

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ & ΔΙΟΙΚΗΣΗΣ

Τι πιστεύετε ότι θα γίνει αν εκτελέσετε το ακόλουθο πρόγραμμα και γιατί;

```
#include <stdio.h>

void main(){
    double trouble = 3.27;
    double *pt;
    *pt = 27.3;
    pt = &trouble;
}
```

segmentation
fault

↑	Διεύθυνση	Μεταβλητή

	401...408	trouble = 3.24

	252...259	pt = (σκουπίδια)

↓		

Δείκτες που δείχνουν σε Δείκτη



Δ.Π.Θ

Δομημένος Προγραμματισμός

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ & ΔΙΟΙΚΗΣΗΣ

```
#include <stdio.h>

int main(){
    int x = 5;
}
```

Διεύθυνση
Μνήμης:



Μεταβλητή:

x

Δείκτες που δείχνουν σε Δείκτη



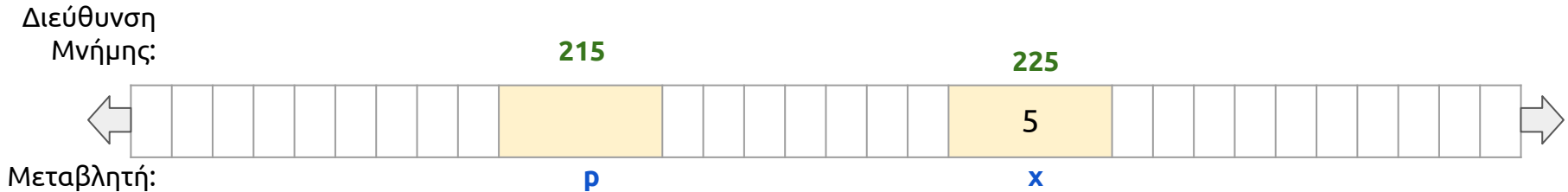
Δ.Π.Θ

Δομημένος Προγραμματισμός

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ & ΔΙΟΙΚΗΣΗΣ

```
#include <stdio.h>

int main(){
    int x = 5;
    int *p;
}
```



Δείκτες που δείχνουν σε Δείκτη



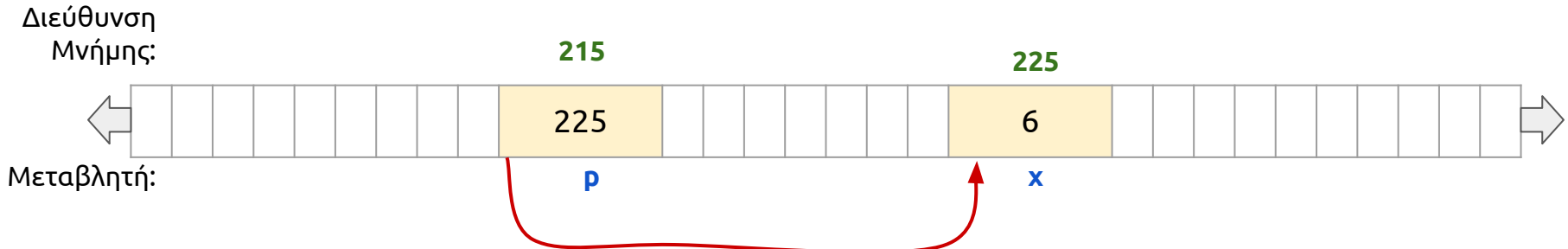
Δ.Π.Θ

Δομημένος Προγραμματισμός

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ & ΔΙΟΙΚΗΣΗΣ

```
#include <stdio.h>

int main(){
    int x = 5;
    int *p;
    p = &x;
    *p = 6;
}
```



Δείκτες που δείχνουν σε Δείκτη



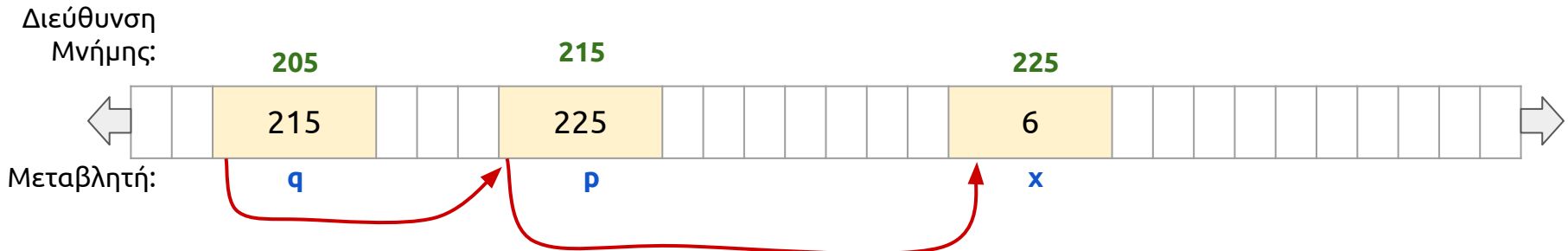
Δ.Π.Θ

Δομημένος Προγραμματισμός

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ & ΔΙΟΙΚΗΣΗΣ

```
#include <stdio.h>

int main(){
    int x = 5;
    int *p;
    p = &x;
    *p = 6;
    int **q = &p;
}
```



Δείκτες που δείχνουν σε Δείκτη



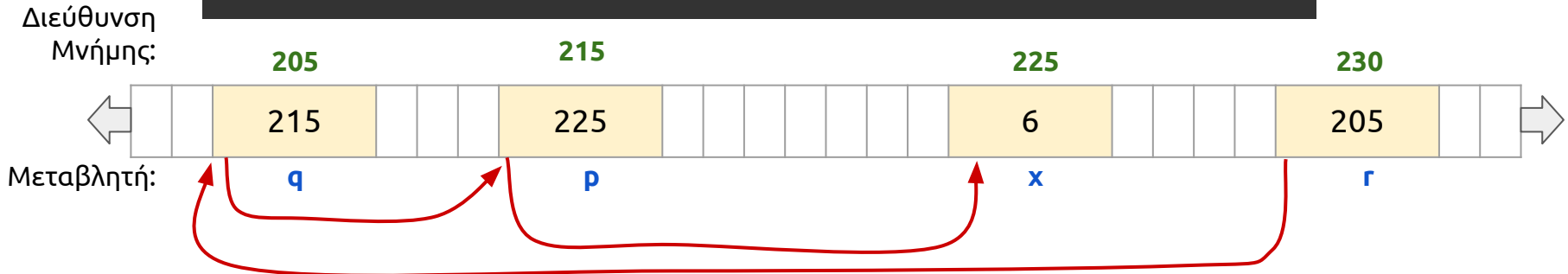
Δ.Π.Θ

Δομημένος Προγραμματισμός

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ & ΔΙΟΙΚΗΣΗΣ

```
#include <stdio.h>

int main(){
    int x = 5;
    int *p;
    p = &a;
    *p = 6;
    int **q = &p;
    int ***r = &q;
}
```



Δείκτες που δείχνουν σε Δείκτη



Δ.Π.Θ

Δομημένος Προγραμματισμός

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ & ΔΙΟΙΚΗΣΗΣ

```
#include <stdio.h>

int main(){
    int x = 5;
    int *p;
    p = &a;
    *p = 6;
    int **q = &p;
    int ***r = &q;
```

```
printf("%d\n", *p); // ???

}
```

Διεύθυνση
Μνήμης:

205

215

225

230

215

225

6

205

q

p

x

r

Μεταβλητή:



Δείκτες που δείχνουν σε Δείκτη



Δ.Π.Θ

Δομημένος Προγραμματισμός

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ & ΔΙΟΙΚΗΣΗΣ

```
#include <stdio.h>

int main(){
    int x = 5;
    int *p;
    p = &a;
    *p = 6;
    int **q = &p;
    int ***r = &q;
```

```
printf("%d\n", *p); // 6

}
```

Διεύθυνση
Μνήμης:

205

215

225

230

215

225

6

205

q

p

x

r

Μεταβλητή:



Δείκτες που δείχνουν σε Δείκτη



Δ.Π.Θ

Δομημένος Προγραμματισμός

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ & ΔΙΟΙΚΗΣΗΣ

```
#include <stdio.h>

int main(){
    int x = 5;
    int *p;
    p = &a;
    *p = 6;
    int **q = &p;
    int ***r = &q;
```

```
printf("%d\n", *p); // 6
printf("%d\n", *q); // ???

}
```

Διεύθυνση
Μνήμης:

205

215

225

230

215

225

6

205

q

p

x

r

Μεταβλητή:



Δείκτες που δείχνουν σε Δείκτη



Δ.Π.Θ

Δομημένος Προγραμματισμός

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ & ΔΙΟΙΚΗΣΗΣ

```
#include <stdio.h>

int main(){
    int x = 5;
    int *p;
    p = &a;
    *p = 6;
    int **q = &p;
    int ***r = &q;
```

```
printf("%d\n", *p); // 6
printf("%d\n", *q); // 225

}
```

Διεύθυνση
Μνήμης:

205

215

225

230

215

225

6

205

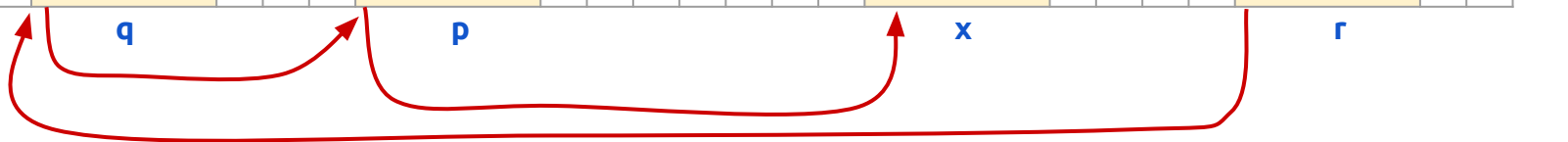
q

p

x

r

Μεταβλητή:



Δείκτες που δείχνουν σε Δείκτη



Δ.Π.Θ

Δομημένος Προγραμματισμός

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ & ΔΙΟΙΚΗΣΗΣ

```
#include <stdio.h>

int main(){
    int x = 5;
    int *p;
    p = &a;
    *p = 6;
    int **q = &p;
    int ***r = &q;
```

```
printf("%d\n", *p); // 6
printf("%d\n", *q); // 225
printf("%d\n", **q); // ???

}
```

Διεύθυνση
Μνήμης:

205

215

225

230

215

225

6

205

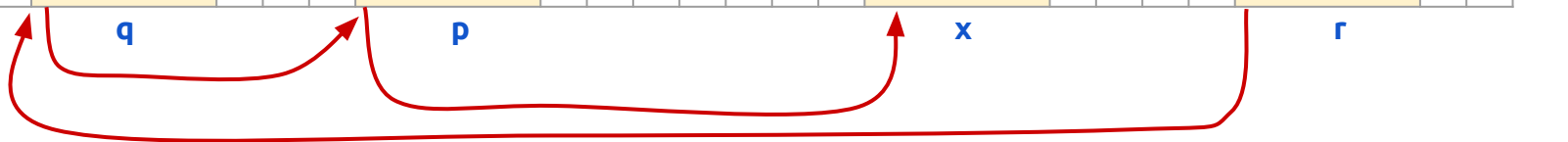
q

p

x

r

Μεταβλητή:



Δείκτες που δείχνουν σε Δείκτη



Δ.Π.Θ

Δομημένος Προγραμματισμός

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ & ΔΙΟΙΚΗΣΗΣ

```
#include <stdio.h>

int main(){
    int x = 5;
    int *p;
    p = &a;
    *p = 6;
    int **q = &p;
    int ***r = &q;
```

```
printf("%d\n", *p); // 6
printf("%d\n", *q); // 225
printf("%d\n", **q); // 6
}
```

Διεύθυνση
Μνήμης:

205

215

225

230

215

225

6

205

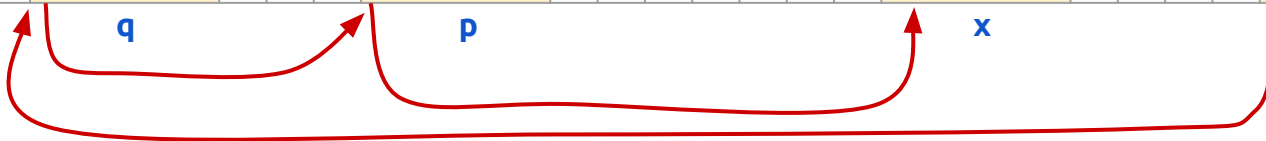
q

p

x

r

Μεταβλητή:



Δείκτες που δείχνουν σε Δείκτη



Δ.Π.Θ

Δομημένος Προγραμματισμός

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ & ΔΙΟΙΚΗΣΗΣ

```
#include <stdio.h>

int main(){
    int x = 5;
    int *p;
    p = &a;
    *p = 6;
    int **q = &p;
    int ***r = &q;
```

```
printf("%d\n", *p); // 6
printf("%d\n", *q); // 225
printf("%d\n", **q); // 6
printf("%d\n", ***r); // ???

}
```

Διεύθυνση
Μνήμης:

205

215

225

230

215

225

6

205

q

p

x

r

Μεταβλητή:



Δείκτες που δείχνουν σε Δείκτη



Δ.Π.Θ

Δομημένος Προγραμματισμός

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ & ΔΙΟΙΚΗΣΗΣ

```
#include <stdio.h>

int main(){
    int x = 5;
    int *p;
    p = &a;
    *p = 6;
    int **q = &p;
    int ***r = &q;
```

```
printf("%d\n", *p); // 6
printf("%d\n", *q); // 225
printf("%d\n", **q); // 6
printf("%d\n", **r); // 225

}
```

Διεύθυνση
Μνήμης:

205

215

225

230

215

225

6

205

q

p

x

r

Μεταβλητή:



Δείκτες που δείχνουν σε Δείκτη



Δ.Π.Θ

Δομημένος Προγραμματισμός

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ & ΔΙΟΙΚΗΣΗΣ

```
#include <stdio.h>

int main(){
    int x = 5;
    int *p;
    p = &a;
    *p = 6;
    int **q = &p;
    int ***r = &q;
```

```
printf("%d\n", *p); // 6
printf("%d\n", *q); // 225
printf("%d\n", **q); // 6
printf("%d\n", **r); // 225
printf("%d\n", ***r); // ???

}
```

Διεύθυνση
Μνήμης:

205

215

225

230

215

225

6

205

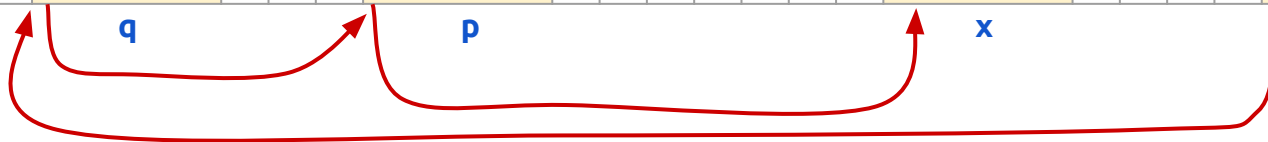
q

p

x

r

Μεταβλητή:



Δείκτες που δείχνουν σε Δείκτη



Δ.Π.Θ

Δομημένος Προγραμματισμός

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ & ΔΙΟΙΚΗΣΗΣ

```
#include <stdio.h>

int main(){
    int x = 5;
    int *p;
    p = &a;
    *p = 6;
    int **q = &p;
    int ***r = &q;
```

```
printf("%d\n", *p); // 6
printf("%d\n", *q); // 225
printf("%d\n", **q); // 6
printf("%d\n", **r); // 225
printf("%d\n", ***r); // 6
}
```

Διεύθυνση
Μνήμης:

205

215

225

230

215

225

6

205

q

p

x

r

Μεταβλητή:



Δείκτες που δείχνουν σε Δείκτη



Δ.Π.Θ

Δομημένος Προγραμματισμός

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ & ΔΙΟΙΚΗΣΗΣ

```
#include <stdio.h>

int main(){
    int x = 5;
    int *p;
    p = &a;
    *p = 6;
    int **q = &p;
    int ***r = &q;
```

```
printf("%d\n", *p); // 6
printf("%d\n", *q); // 225
printf("%d\n", **q); // 6
printf("%d\n", **r); // 225
printf("%d\n", ***r); // 6
***r = 10;
printf("%d\n", x); // ???

}
```

Διεύθυνση
Μνήμης:

205

215

225

230

215

225

6

205

q

p

x

r

Μεταβλητή:



Δείκτες που δείχνουν σε Δείκτη



Δ.Π.Θ

Δομημένος Προγραμματισμός

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ & ΔΙΟΙΚΗΣΗΣ

```
#include <stdio.h>

int main(){
    int x = 5;
    int *p;
    p = &a;
    *p = 6;
    int **q = &p;
    int ***r = &q;
```

```
printf("%d\n", *p); // 6
printf("%d\n", *q); // 225
printf("%d\n", **q); // 6
printf("%d\n", **r); // 225
printf("%d\n", ***r); // 6
***r = 10;
printf("%d\n", x); // 10
}
```

Διεύθυνση
Μνήμης:

205

215

225

230



215

225

10

205

q

p

x

r

Μεταβλητή:



Δείκτες που δείχνουν σε Δείκτη



Δ.Π.Θ

Δομημένος Προγραμματισμός

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ & ΔΙΟΙΚΗΣΗΣ

```
#include <stdio.h>

int main(){
    int x = 5;
    int *p;
    p = &a;
    *p = 6;
    int **q = &p;
    int ***r = &q;
```

```
printf("%d\n", *p); // 6
printf("%d\n", *q); // 225
printf("%d\n", **q); // 6
printf("%d\n", **r); // 225
printf("%d\n", ***r); // 6
***r = 10;
printf("%d\n", x); // 10
**q = *p + 2;
printf("%d\n", x); // ???
}
```

Διεύθυνση
Μνήμης:

205

215

225

230

215

225

10

205

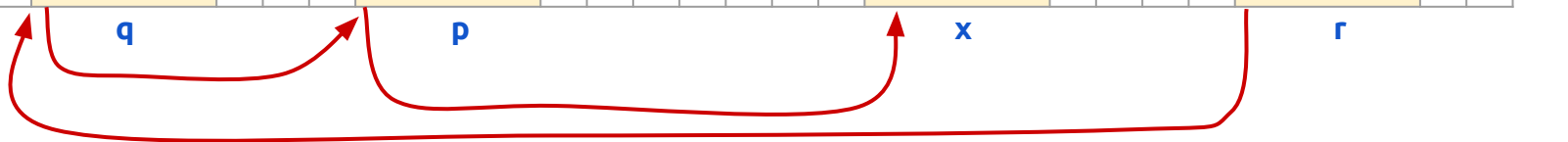
q

p

x

r

Μεταβλητή:



Δείκτες που δείχνουν σε Δείκτη



Δ.Π.Θ

Δομημένος Προγραμματισμός

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ & ΔΙΟΙΚΗΣΗΣ

```
#include <stdio.h>

int main(){
    int x = 5;
    int *p;
    p = &a;
    *p = 6;
    int **q = &p;
    int ***r = &q;
```

```
printf("%d\n", *p); // 6
printf("%d\n", *q); // 225
printf("%d\n", **q); // 6
printf("%d\n", **r); // 225
printf("%d\n", ***r); // 6
***r = 10;
printf("%d\n", x); // 10
**q = *p + 2;
printf("%d\n", x); // 12
}
```

Διεύθυνση
Μνήμης:

205

215

225

230

215

225

12

205

q

p

x

r

Μεταβλητή:

