

Εισαγωγή στην Πληροφορική & στον Προγραμματισμό

Αρχές Προγραμματισμού Η/Υ (με τη γλώσσα C) – **Εργαστηριακή online συνάντηση**

Διάλεξη #4

Παρασκευή, 26 Μαρτίου 2022

Παναγιώτης Παύλου

c-programming-22@allos.gr

Ανακεφαλαίωση

Με μία ματιά να θυμηθούμε τη θεωρία

Μεταβολές

$a = *x;$
 $x++;$
 $a = x++;$
 $a = 7$
 $b = ++x;$
 $b = 8$
 $x = x + a;$
 $x = 1$

Για τους αριθμητικούς τελεστές οι εκφράσεις:

someVariableName = someVariableName + (change);

γράφονται και

someVariableName += change;

άρα για τους αριθμητικούς τελεστές έχουμε:

+= -= *=
/= %=

Σημειώστε ότι το παρακάτω δεν γράφεται συνοπτικά:

x = a - x;

$a = b = c$
 $a = c$

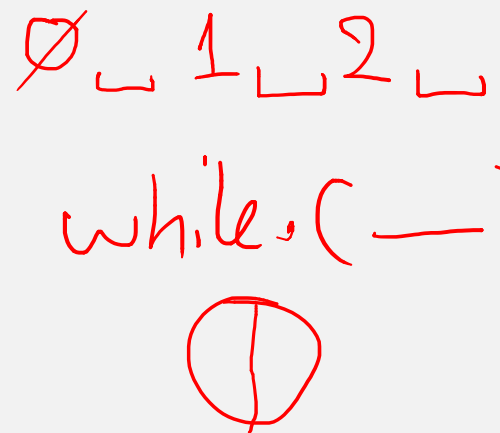
Ειδικά για πρόσθεση και την αφαίρεση, όταν change = 1 γράφω:

$x++;$ ή $++x;$ και $x--;$ ή $--x;$

Βρόχοι - Loops



```
int i = 0;  
while (i < 10) {  
    printf("%d ", i);  
    i++;  
}
```



Αρχικοποίηση
Συνθήκη
Μεταβολή

```
for (int i = 0; i < 10; i++) {  
    printf("%d ", i);  
}
```

int x = 2, y = 3;

Οι επαναλήψεις μπορεί να είναι από **καμία** μέχρι **άπειρες**.

Πιο σπάνια σχήματα ελέγχου ροής

$a + x ==$ $\begin{matrix} 3 & 17 \\ \downarrow & \\ \end{matrix}$

```
int i = 0;
do {
    int j = 0;
    printf("%d ", i);
    i++;
} while (i <= 10);
```

Handwritten annotations: A red bracket groups the `do` block. A red circle highlights the closing brace of the `do` block. A red circle highlights the semicolon at the end of the `while` loop. A red circle highlights the semicolon at the end of the `while` loop. A red circle highlights the semicolon at the end of the `while` loop. A red circle highlights the semicolon at the end of the `while` loop.

```
switch (base) {
    case 10:
        // δεν απαιτείται κάτι
        break;
    case 8:
        // μετατροπή σε βάση 16
    case 16:
        // μετατροπή σε βάση 10
        break;
    default:
        // εμφάνιση σφάλματος
        break;
}
```

Handwritten annotations: A red circle highlights the `10` in the first case. A red circle highlights the `8` in the second case. A red circle highlights the `16` in the third case. A red circle highlights the `default` label. A red circle highlights the `break` in the default case. A red circle highlights the closing brace of the `switch` block. A red circle highlights the `default` label. A red circle highlights the `break` in the default case. A red circle highlights the `break` in the default case. A red circle highlights the `break` in the default case.

if (A)

~~10 20 30~~

x < 10

x < 20

x < 30

if (B)

else

(x < 1 || y > 2) && (x > -5)

&&

!(a < b)

if (A && ! B)

Εφαρμογές

Ας εφαρμόσουμε επιτέλους τη θεωρία

if (—)
else if (—)

	T	F
A	M	x
B	M	x

Πρακτική 1



Δημιουργήστε τα ακόλουθα σχήματα στην κονσόλα.

Πρώτα το:

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

A

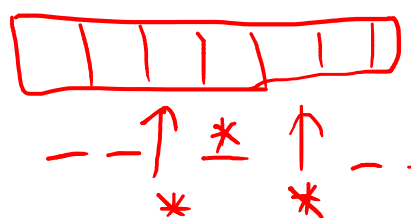
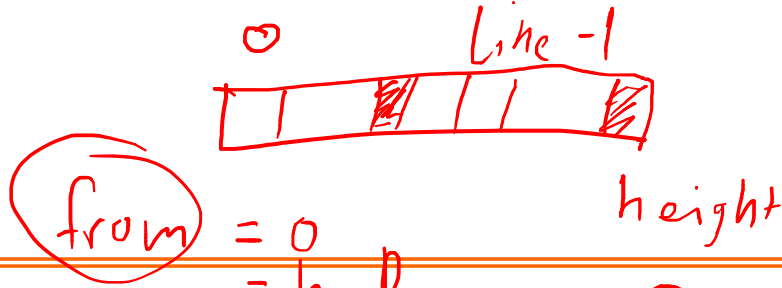
μέσω μιας συνάρτησης

```
void drawTriangle(int height)
```

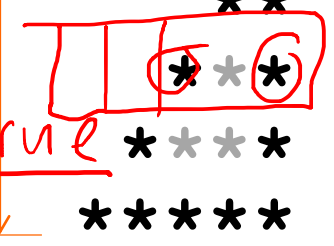
και ύστερα το:

```
□□□*
□□**
□***
****
```

$$\text{spc} + \text{ast} = \text{height} - \text{ast}$$



$$\text{to} = \text{h} - 1$$



ROW	Πόσα *	Πόσα _
0	1	4
1	2	3
2	3	2
3	4	1
4	5	0

μέσω επέκτασης της που να παίρνει σαν 2^η παράμετρο το bool right

Επέκταση της συνάρτησης με 3^η παράμετρο bool outline που να σχεδιάζει το περίγραμμα του τριγώνου.

Βιβλιοθήκη του μαθήματος

smLib : Η βιβλιοθήκη που θα μας βοηθά στο μάθημα

Τι είναι

Η smLib είναι μία βιβλιοθήκη που δημιουργήθηκε για τις ανάγκες του μαθήματος, ώστε να μας επιτρέψει να μάθουμε κάθε τι στην ώρα του.

Η C από τη φύση της, όντας κοντά στον επεξεργαστή, απαιτεί αρκετές γνώσεις για να γίνουν ακόμα και απλά πράγματα. Χωρίς αυτές να είναι δύσκολες, δεν μπορούμε να τις μάθουμε στιγμιαία, αλλά σταδιακά. Σε αυτό ακριβώς συμβάλλει η βιβλιοθήκη αυτή.

Το πρόθεμα sm είναι τα αρχικά της Σχολής Μηχανολόγων. Ό,τι βλέπετε με αυτό το πρόθεμα, πχ συνάρτηση ή τύπο δεδομένων, έχει δημιουργηθεί για τις ανάγκες του μαθήματος και μέχρι το τέλος του εξαμήνου δεν θα το χρειαζόμαστε, αφού θα έχουμε καλύψει όλη τη βασική λειτουργία της γλώσσας.

Τύπος δεδομένων smString

Είναι ένας τύπος δεδομένων που παριστάνει ένα κείμενο, όπως αυτά που βλέπουμε σε διπλά εισαγωγικά.

Μέχρι να μάθουμε να διαχειριζόμαστε τα κείμενα με τον τρόπο της C μας επιτρέπει να χρησιμοποιήσουμε κείμενα για τις απλές λειτουργίες τους.

Δηλώνω πχ μια μεταβλητή που μπορεί να αποθηκεύσει ένα κείμενο:

```
smString text = "Kalimera";
```

και ανάλογα δηλώνω και μία συνάρτηση που επιστρέφει κείμενο ή δέχεται ένα κείμενο ως παράμετρό της.

smReadInt – εισαγωγή δεδομένων

Η `smReadInt` μας επιτρέπει να εμφανίζουμε ένα μήνυμα στον χρήστη και κατόπιν του επιτρέπει να μας δώσει έναν αριθμό από το πληκτρολόγιο (αφού πατήσει `enter`) το οποίο και μας τον επιστρέφει. Ο αριθμός εισάγεται από τον χρήστη κάτω από τη γραμμή του μηνύματος.

Η σύνταξη της εντολής είναι:

```
int x = smReadInt("μήνυμα προς τον χρήστη");
```

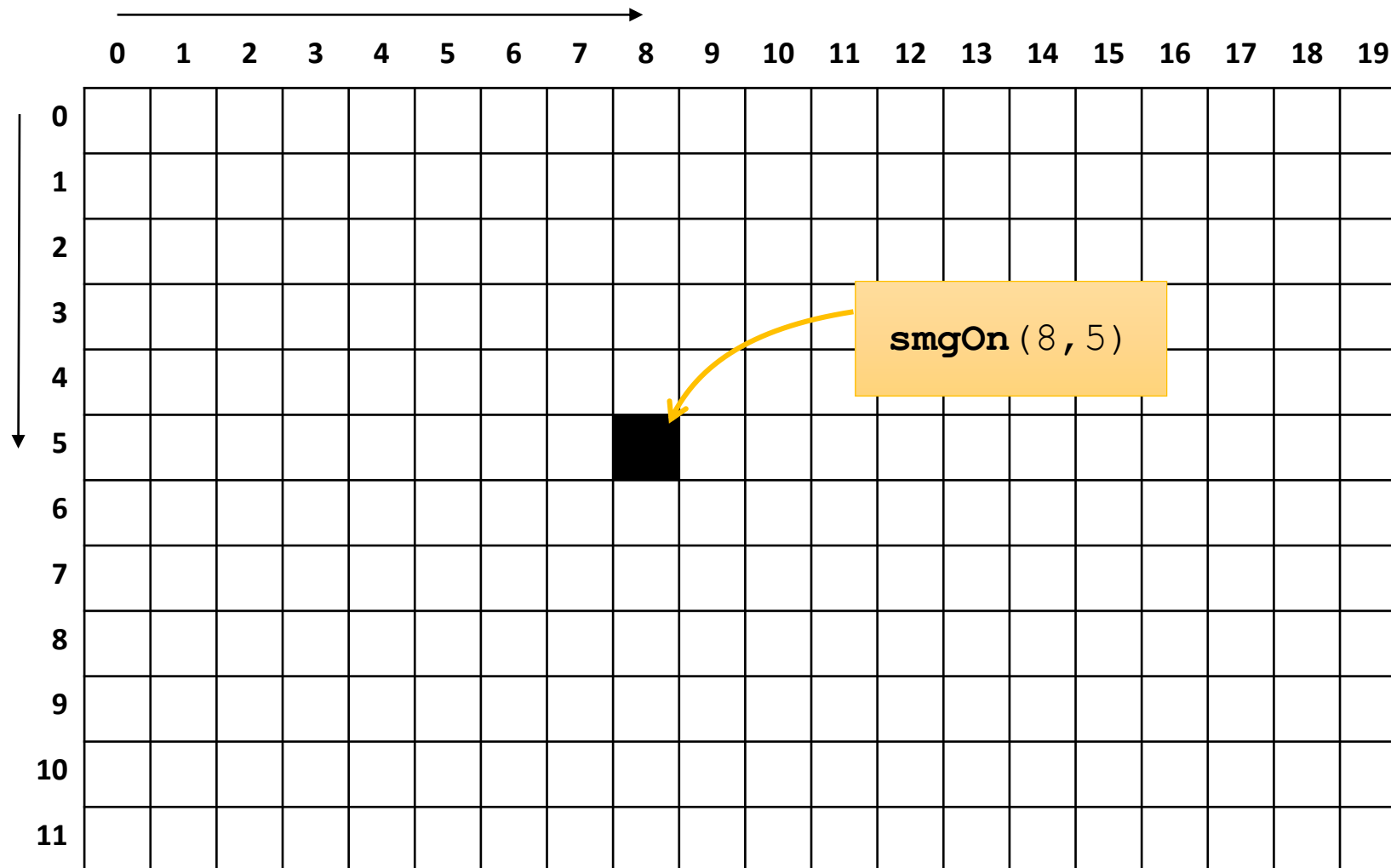
Εάν ο χρήστης δεν δώσει όντως αριθμό η εντολή επιμένει μέχρι να πάρει μια απάντηση.

Εκπαιδευτικό πλέγμα / Grid

Το πλέγμα είναι μία διάταξη πίνακα στην οποία μπορούμε να «ανάβουμε» και να «σβήνουμε» τα στοιχεία του. Οι συντεταγμένες μετρώνε από πάνω αριστερά, όπου βρίσκεται το κελί 0,0. Επίσης αυξάνονται προς τα δεξιά και προς τα κάτω, δηλαδή η κατακόρυφη διεύθυνση είναι ανάποδη από το κλασσικό x,y σύστημα συντεταγμένων. Εξορισμού το πλέγμα έχει διαστάσεις 20x12, όμως μπορείτε να το αλλάξετε αυτό πριν το χρησιμοποιήσετε για πρώτη φορά σε κάθε κώδικα.

Οπτική αναπαράσταση grid

Οπτικά το πλέγμα είναι όπως ο παρακάτω πίνακας.



Εντολές για το grid

Για να ορίσετε τις **διαστάσεις** του πλέγματος καλείτε την συνάρτηση

smGrid(πλατος, ύψος)

Συνήθως θα είναι η 1^η εντολή από αυτές που σχετίζονται με το grid. Εάν την εκτελέσετε 2^η φορά αργότερα, τότε αλλάζει το μέγεθος του grid, αλλά σβήνει όλο το προηγούμενο περιεχόμενο.

Εάν δεν καλέσετε τη συνάρτηση αυτή, τότε θεωρείται πλέγμα 20x12.

Για να **ανάψετε** ένα κελί καλείτε την

smgOn(x, y)

Όπου x και y είναι οι οριζόντια και κατακόρυφη συντεταγμένη αντίστοιχα. Η αρίθμηση ξεκινά από το 0.

Εντολές για το grid

Για να δείτε την κατάσταση ενός κελιού καλέστε την

smgTest (x, y)

η οποία επιστρέφει αληθές εάν το κελί είναι αναμένο ή ψευδές εάν είναι σβηστό.

Επίσης για να δείτε τις διαστάσεις του grid, χρησιμοποιήστε τις

smgWidth ()

και

smgHeight ()

που επιστρέφουν την αντίστοιχη διάσταση.

Ερωτήσεις?

- Διαβάστε τις σημειώσεις, διαβάστε τις διαφάνειες και δείτε τα videos **πριν** ρωτήσετε
- **Συμβουλευτείτε** τη σελίδα ερωταποκρίσεων του μαθήματος
<https://qna.c-programming.allos.gr>
- **Στείλτε** τις ερωτήσεις σας πριν και μετά το μάθημα στο
c-programming-22@allos.gr
- Εάν έχετε **πρόβλημα** με κάποιο κώδικα στείλτε μαζί τον κώδικα και τα μηνύματα λάθους από το CLI ως κείμενα με copy/paste. Εάν θεωρείτε ότι επιπλέον βοηθά και ένα στιγμιότυπο οθόνης, είναι καλοδεχούμενο.
- Επαναλαμβάνουμε : Μην στείλετε ποτέ κώδικα ως εικόνα μας είναι παντελώς άχρηστος!



Πρακτική 2 – εκτός πλάνου



Γράψτε μια συνάρτηση που να εκτυπώνει την ημερομηνία στη μορφή:

Friday, 26 March 2021

switch / case

0 1 2

Για τις ανάγκες του παραδείγματος υποθέτουμε ότι το έτος θα είναι πάντα το 2021. Επίσης δίνεται ότι η 1/1/2021 ήταν Παρασκευή.

```
void printEngDate(int day, int month, int year);
```

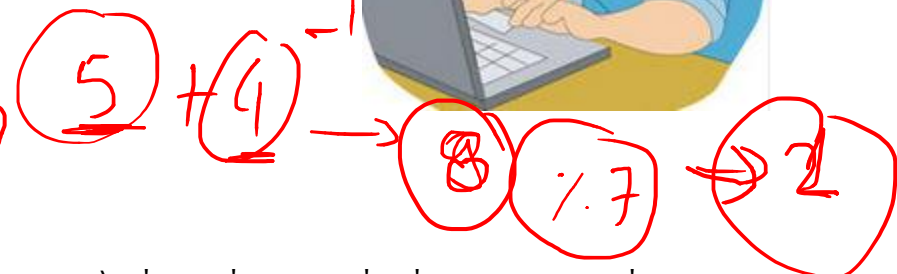
Θα σας βοηθήσει πολύ να δημιουργήσετε συναρτήσεις που να αναλαμβάνουν τα επιμέρους προβλήματα.

Πρακτική 2 – μερικές σκέψεις

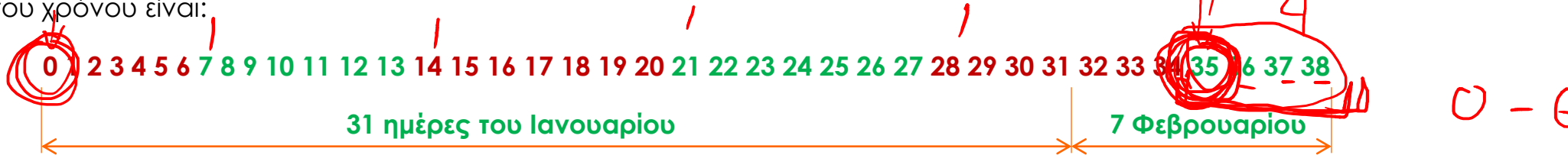


Κοιτώντας την μορφή της ημ/νίας βλέπουμε ότι χρειάζεται να έχουμε:

- το όνομα της ημέρας – κωδικοποίηση ημερών εβδομάδας **0=Κυριακή – 6=Σάββατο**
- το όνομα του μήνα – κωδικοποίηση μηνών **1=Ιανουάριος – 12=Δεκέμβριος**



Επίσης για να βρούμε την ημέρα της εβδομάδας θα χρειαστούμε να υπολογίσουμε τις συνολικές ημέρες από κάποια γνωστή ημερομηνία (εδώ δίνεται η 1/1/2021) και να δούμε πόσες ημέρες περισσεύουν από την τελευταία ολόκληρη εβδομάδα, π.χ. για την 39^η ημέρα του χρόνου είναι:



αφού συμπληρώνονται 5 εβδομάδες και περισσεύουν 4 ημέρες, από την εβδομάδα αυτή την τελευταία το 35 θα είναι Τετάρτη, άρα το 38 Σάββατο. Αυτό «μυρίζει» υπόλοιπο διαίρεσης και δικαιολογεί την επιλογή των αριθμών 0-6 για την κωδικοποίηση των ημερών.

Επίσης πρέπει να υπολογιστούν και οι ημέρες που έχουν περάσει από την 1/1/2021. Αλλά για να γίνει αυτό πρέπει να γνωρίζουμε πόσες ημέρες έχει ο κάθε μήνας.

Αυτά τα υπογραμμισμένα επί μέρους προβλήματα μπορούν να γίνουν ξεχωριστές συναρτήσεις.

ΠΕΡΙΣΣΟΤΕΡΗ ΕΞΑΣΚΗΣΗ
8, 9, 2025

2021 < 2025

Για εξάσκηση δική σας, **χωρίς** όμως να αποτελεί εργασία προς παράδοση, μπορείτε επεκτείνετε τον κώδικα για να λειτουργεί και για άλλα έτη. Θυμηθείτε ότι ήδη έχουμε συναντήσει τη συνάρτηση που υπολογίζει εάν ένα έτος είναι δίσεκτο.

$$365 + 365 + 365 + 366 + 31 + 28 = 31 + 30 + 31 + 31 + 8$$