

Επανάληψη και Παλιά θέματα εξετάσεων

Θέμα 1

Ποιά είναι η έξοδος του ακόλουθου προγράμματος;
Εξηγήστε την απάντησή σας.

```
#include <stdio.h>

#define MF1(X)    ((X > 0) ? X : -(X))
#define MF2(X)    (X > 0) ? X : -X

int main()
{
    int  a=5, b=10;

    printf("MF1=%d, MF2=%d\n",MF1(a-b),MF2(a-b));

    return 0;
}
```

Θέμα 2

α) Να γράψετε μια συνάρτηση με όνομα **isvowel** που να δέχεται στην είσοδό της έναν χαρακτήρα και να επιστρέφει την τιμή 1 αν αυτός αντιστοιχεί σε φωνήεν. Σε αντίθετη περίπτωση, να επιστρέφει την τιμή 0.

β) Να γράψετε ένα πρόγραμμα που να προτρέπει τον χρήστη να πληκτρολογήσει μια γραμμή με αλφαριθμητικούς χαρακτήρες και να εμφανίζει στην οθόνη το πλήθος των φωνηέντων. Το πρόγραμμα να χρησιμοποιεί τη συνάρτηση **isvowel** του προηγούμενου ερωτήματος.

Θέμα 3

Να εξηγήσετε τι κάνει η παρακάτω συνάρτηση και ποιές τιμές επιστρέφουν οι κλήσεις **function(4)** και **function(-2)**;

```
int function(int n)
{
    if (n == 0 || n == 1)
        return(1);
    else if (n > 1)
        return(n * function(n-1));
    else
        return(0);
}
```

Θέμα 4

Να εξηγήσετε γραμμή προς γραμμή τι κάνει το παρακάτω πρόγραμμα.

```
#include <stdio.h>
int main()
{
    int i,dim;
    float *MAT;
    FILE *fp;
    if((fp = fopen("pinakas.dat","r")) != NULL)
    {
        fscanf(fp,"%d",&dim);
        MAT = (float *) malloc(dim*sizeof(float));
        for(i=0; i<dim; i++)
            fscanf(fp,"%f",MAT+i);
        fclose(fp);
        for(i=0; i<dim; i++)
            printf("MAT[%d] = %f\n",i,* (MAT+i));
        free(MAT);
    }
    else exit(-1);
    return 0;
}
```

Θέμα 5

α) Να γράψετε μια συνάρτηση με δύο ορίσματα εισόδου: α) μια συμβολοσειρά και β) έναν χαρακτήρα. Η συνάρτηση θα απαλείφει όλες τις (πιθανές) εμφανίσεις του χαρακτήρα μέσα στη συμβολοσειρά και θα επιστρέφει μια ακέραια τιμή που αντιστοιχεί στο πλήθος των εμφανίσεων του χαρακτήρα μέσα στη συμβολοσειρά.

β) Να γράψετε ένα πρόγραμμα που να προτρέπει τον χρήστη να πληκτρολογήσει μια συμβολοσειρά (οι χαρακτήρες μπορεί να είναι αλφαριθμητικοί ή και λευκοί, δηλαδή, κενό ' ' ή tab '\t' αλλά όχι και αλλαγή γραμμής '\n') και να εμφανίζει στην οθόνη τη συμβολοσειρά από την οποία θα έχουν αφαιρεθεί όλες οι εμφανίσεις του χαρακτήρα 'a'. Το πρόγραμμα να χρησιμοποιεί τη συνάρτηση του προηγούμενου ερωτήματος.

Θέμα 6

Να γράψετε ένα πρόγραμμα που να δημιουργεί ένα ASCII αρχείο (π.χ. "file1.txt") στο οποίο θα αποθηκεύονται, σε δύο γραμμές, το ονοματεπώνυμο και ο αριθμός μητρώου του χρήστη.

Θέμα 7

Έστω ότι στο ASCII αρχείο "names.txt" έχουμε αποθηκεύσει N ονοματεπώνυμα φοιτητών (ένα σε κάθε γραμμή). Το πλήθος N είναι αποθηκευμένο στην πρώτη γραμμή του αρχείου (βλ. παράδειγμα, για N = 3). Να γράψετε πρόγραμμα που να ανοίγει το αρχείο, να διαβάζει το N, να δεσμεύει μνήμη για πίνακα N συμβολοσειρών, να διαβάζει τα ονοματεπώνυμα και να τα αποθηκεύει στον πίνακα συμβολοσειρών και, τέλος, να τα εμφανίζει και να κλείνει το αρχείο.

```
3
ΓΙΑΝΝΟΥΚΟΥ ΕΥΔΟΣΙΑ
ΚΑΡΑΓΕΩΡΓΗΣ ΚΙΜΩΝ
ΚΑΡΑΦΩΤΗ ΙΩΑΝΝΑ
```

Παράδειγμα αρχείου "names.txt"

Θέμα 8

Να γίνει προσομοίωση με το χέρι του παρακάτω βρόχου;

```
int i, j;

for(i=0, j=10; (i < j) || (i = j-2); i++, --j)
    printf("i = %d, j = %d\n", i, j);

printf("i = %d, j = %d\n", i, j);
```

Θέμα 9

Να γραφεί συνάρτηση με όνομα **median()** που να δέχεται στην είσοδό της έναν πίνακα ακεραίων μαζί με το πλήθος τους, να τους ταξινομεί σε αύξουσα διάταξη και να επιστρέφει τον *διάμεσο* αριθμό. Αν το πλήθος των αριθμών είναι περιπτό ($N=2k+1$) τότε ο διάμεσος θα είναι στην $(k+1)$ θέση ενώ αν είναι άρτιο ($N=2k$) στη θέση k .

αρχικοί αριθμοί *sorting* σε αύξουσα διάταξη

Παράδειγμα: 29 6 9 20 2 20 23 --> 2 6 9 20 20 23 29

7 9 1 18 9 9 --> 1 7 9 9 18

μεσαίος αριθμός

Θέματα 10 & 11

Στο αρχείο blocks.txt υπάρχει η καταγραφή από τα τουβλάκια (blocks) ενός παιδικού παιχνιδιού. Τα τουβλάκια μπορεί να έχουν σχήμα τετραγωνικό (square), κυκλικό (circle) ή ορθογώνιο (rectangle). Το χρώμα τους μπορεί να είναι κόκκινο (red), πράσινο (green) ή μπλε (blue). Το κάθε τουβλάκι μπορεί να έχει το δικό του βάρος (weight) και το δικό του ύψος (height). Το πλήθος των μπλοκ αναφέρεται στην πρώτη γραμμή του αρχείου. Στη δεύτερη γραμμή εμφανίζουμε μνημονικά ονόματα για το σχήμα (shape), χρώμα, βάρος και ύψος του κάθε μπλοκ. Στη συνέχεια σε κάθε επόμενη γραμμή του αρχείου εμφανίζονται τα στοιχεία του κάθε μπλοκ όπως στο παρακάτω παράδειγμα:

Plithos blocks: 7			
shape	color	weight (gr)	height (cm)
square	red	9.7	1.6
rectangle	blue	13.6	1.6
circle	blue	9.0	1.5
circle	green	9.9	1.8
rectangle	red	15.1	2.1
square	red	10.2	1.9
square	red	10.6	1.8

α) Να ορίσετε τη δομή block (τουβλάκι) με τέσσερα πεδία για τα εξής: i) shape για το σχήμα του μπλοκ (π.χ. square, circle, rectangle), ii) color για το χρώμα του μπλοκ (π.χ. red, blue, green), iii) weight για το βάρος του (πραγματικός αριθμός) και iv) height για το ύψος του (πραγματικός αριθμός).

...(συνέχεια)

β) Να γράψετε ένα πρόγραμμα που να διαβάζει το πλήθος των μπλοκ από την πρώτη γραμμή του αρχείου και να δεσμεύει δυναμικά μνήμη για πίνακα από δομές τύπου **block** στον οποίο θα αποθηκευθούν τα στοιχεία των μπλοκ του αρχείου. Τόσο ο δείκτης στη δεσμευμένη μνήμη όσο και το πλήθος των στοιχείων του πίνακα να ορισθούν ως καθολικές (global) μεταβλητές. Επίσης, το πρόγραμμα θα εμφανίζει στην οθόνη τα μπλοκ που διαβάσαμε (με τον ίδιο τρόπο που εμφανίζονται στο αρχείο) ταξινομημένα σε φθίνουσα διάταξη ως προς το βάρος τους με χρήση της συνάρτησης `sortblocks` που περιγράφεται στη συνέχεια.

γ) Γράψτε τη συνάρτηση `sortblocks` που δέχεται σαν παράμετρο μια από τις συμβολοσειρές "weight" ή "height" και εμφανίζει στην οθόνη, με τον ίδιο τρόπο που εμφανίζονται και στο αρχείο, όλα τα στοιχεία των μπλοκ που διάβασε το κυρίως πρόγραμμα του προηγούμενου θέματος ταξινομημένα σε φθίνουσα διάταξη ως προς την τιμή του αντίστοιχου πεδίου (weight ή height). Η συνάρτηση θα χρησιμοποιεί τις καθολικές μεταβλητές (δείκτη και διάσταση πίνακα) του κυρίως προγράμματος.

Θέμα 12

α) Ποια είναι η έξοδος του παρακάτω προγράμματος υποθέτοντας byte addressable σύστημα;

```
#include <stdio.h>

struct foo {
    int c;
    char s1[20];
    char *s2;
    double x;
};

int main()
{
    short int a[10],*b;
    struct foo v;

    printf("%d\n%d\n%d\n",sizeof a, sizeof b, sizeof v);
    return 0;
}
```

... ΣΥΝΕΧΕΙΑ

β) Ποια η έξοδος του παρακάτω προγράμματος;

```
#include <stdio.h>
int func(int);

int main()
{
    int i, a=5;

    for(i=-2; i<6; i+=3)
        { a = func(a+i)%5; printf("a = %d\n",a); }
    return 0;
}

int func(int x)
{
    static int y=1;

    y += x;
    return y;
}
```

... ΣΥΝΕΧΕΙΑ

γ) Τι θα εμφανίσει το παρακάτω πρόγραμμα;

```
#include <stdio.h>
void func1(void), func2(void);
int func3(int, int);

int x, y, a;

int main()
{
    int x=1, y=0;

    func1();
    a = func3(x, y);
    func2();

    printf("a = %d\n", a);
    return(0);
}

void func1(void)
{
    x = 10;
    y = 5;
}

void func2(void)
{
    a = 0;
    printf("x = %d y = %d\n", x, y);
}

int func3(int i1, int i2)
{
    int diafora;
    diafora = i1 - i2;
    return(diafora);
}
```

Θέμα 13

Να δημιουργήσετε μια συνάρτηση `cleanstr()` ενός ορίσματος η οποία θα δέχεται μια συμβολοσειρά στην είσοδό της, θα απαλείφει όλους τους αριθμητικούς χαρακτήρες και θα επιστρέφει (με τη λέξη κλειδί `return`) έναν `pointer` στη νέα συμβολοσειρά. Στην περίπτωση που δεν υπάρχουν αριθμητικοί χαρακτήρες, θα επιστρέφει την ίδια συμβολοσειρά.

Π.χ. η `cleanstr("cs10001@teiath.gr")` θα επιστρέψει δείκτη στη συμβολοσειρά `"cs@teiath.gr"` ενώ η `cleanstr("sqrt(a^b)")` θα επιστρέψει `"sqrt(a^b)"`.

Θέμα 14

Να γράψετε ένα πρόγραμμα που να ανοίγει το αρχείο `text1.txt` και να εμφανίζει τα περιεχόμενά του στην οθόνη απαλείφοντας τους αριθμητικούς χαρακτήρες. Το πρόγραμμα να χρησιμοποιεί την `cleanstr()`.

Θέμα 15

Να γράψετε ένα πρόγραμμα που θα ανοίγει το αρχείο κειμένου `"C_grades.txt"` στο οποίο είναι αποθηκευμένοι οι βαθμοί (πραγματικοί αριθμοί) για το μάθημα της C και θα τους διαβάξει σε ένα πίνακα. Το πλήθος των βαθμών που θα διαβαστούν (ακέραιος) θα είναι στην πρώτη γραμμή του αρχείου και ο πίνακας θα δημιουργείται με δυναμική δέσμευση μνήμης. Στη συνέχεια, το πρόγραμμα θα υπολογίζει τον μέγιστο, ελάχιστο και μέσο όρο των βαθμών και θα τους εμφανίζει στην οθόνη.

Θέμα 16

Γράψτε μια συνάρτηση με δύο παραμέτρους, τους ακεραίους `pos` και `cap`. Η συνάρτηση θα επιστρέφει το γράμμα που βρίσκεται στην αντίστοιχη θέση `pos` του λατινικού αλφαβήτου και συγκεκριμένα αν το `cap` είναι 1 θα επιστρέφει το κεφαλαίο ενώ αν είναι 0 θα επιστρέφει το αντίστοιχο πεζό γράμμα. Στην περίπτωση που κάποια παράμετρος είναι εκτός ορίων η συνάρτηση θα επιστρέφει το χαρακτήρα `'\0'`. Π.χ.

Παράμετροι		Η συνάρτηση επιστρέφει:
<code>pos</code>	<code>cap</code>	
4	1	D
4	0	d
26	1	Z
27	0	<code>\0</code>
3	2	<code>\0</code>
-4	1	<code>\0</code>