

ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ  
Τμήμα Πληροφορικής και Τηλεπικοινωνιών  
Κ22: Λειτουργικά Συστήματα (Άρτιοι Αριθ. Μητρώου) – Χειμερινό '19

1ο Σετ Ασκήσεων  
Ημερ. Ανακοίνωσης: 9/10  
Ημερ. Υποβολής: 16/10 – Ώρα 15:00 (στην τάξη)

Δώστε σύντομες αλλά περιεκτικές απαντήσεις —μέσα στο πολύ 3-4 δακτυλογραφημένες ή ευανάγνωστα χειρόγραφες σελίδες— σε όλα στα παρακάτω ερωτήματα:

- Πρόβλημα 1 (10 πόντοι):  
Ποια είναι η διαφορά μεταξύ ενός Λειτουργικού Συστήματος για server και ενός workstation;
- Πρόβλημα 2 (15 πόντοι):  
Γιατί υπάρχουν δυο modes εκτέλεσης προγραμμάτων σε ένα Λειτουργικό Σύστημα; Σε τι αποσκοπούν;
- Πρόβλημα 3 (15 πόντοι):  
Υπάρχει κάποια διαφορά μεταξύ ενός trap από ένα hardware interrupt ; Δικαιολογήστε την απάντησή σας.
- Πρόβλημα 4 (20 πόντοι):  
Γιατί το DMA είναι μηχανισμός που είναι εξαιρετικά χρήσιμος σε συσκευές όπως οι μαγνητικοί δίσκοι, οι SSDs και τα οπτικά μέσα (DVDs κλπ).
- Πρόβλημα 5 (20 πόντοι):  
Θεωρήστε το παρακάτω κώδικα:

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

int main(){
    int *pa=(int *)malloc(sizeof(int));
    *pa=1234;
    printf("The value found at *pa is %d \n",*pa);
    {
        int temp1=2345;
        pa= &temp1;
    }
    printf("The value found at *pa is %d \n",*pa);
    return 1;
}
```

Υπάρχει κάποιο σημαντικό πρόβλημα σε αυτό το κώδικα; Αν υπάρχει εξηγήστε πιο είναι και πως μπορεί να διορθωθεί. Αν δεν υπάρχει εξηγήστε γιατί;

- Πρόβλημα 6 (20 πόντοι): Θεωρήστε το παρακάτω πρόγραμμα:

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

void *myfunction(int flag, void *alpha){
    static int mylocal=3;
    int aa[10];

    if (flag==0){
        printf("return the declared table (address of) !\n");
        return aa;
    }
    else if (flag==1){
        printf("change the 2nd function param!\n");
        return alpha;
    }
    else if (flag==2){
        int *myint= malloc(sizeof(int));
        *myint=1245;
        printf("return the locally declared pointer!\n");
        return myint;
    }
    else
    {
        printf("return value of mylocal variable!\n");
        return &mylocal;
    }
}

int main(int argc, char *argv[]){
    int *ii=(int *)malloc(sizeof(int));
    float *yy;

    ii=myfunction(atoi(argv[1]), yy);
    return 1;
}
```

Το πρόγραμμα φαίνεται να περνάει gcc compilation με επιτυχία και να τρέχει ικανοποιητικά για inline parameter τιμές που δίνουμε. Διαπιστώνετε κάποια πιθανά προβλήματα με τον παραπάνω κώδικα;

Αν ναι ποια είναι αυτά τα λάθη και γιατί τα θεωρείτε (σημαντικά) προβλήματα;

Οι συνήθεις κανόνες για αντιγραφή ισχύουν.

1. Οι απαντήσεις είναι δικές σας και ο γραπτός λόγος είναι *προσωπικά δική σας σύνθεση*.
2. Φυσικά και *δεν μπορείτε να κάνετε copy-paste* απαντήσεις.
3. Αν έχετε συμβουλευτεί κάποια άλλη πηγή εκτός από το βιβλίο(-α) σας *θα πρέπει να αναφέρετε με λεπτομερή τρόπο την εν λόγω πηγή αναφοράς σας στο τέλος της άσκησης σας*.

Αν οι απαντήσεις είναι γραμμένες σε ένα text formatter (δηλ. L<sup>A</sup>T<sub>E</sub>X, LibreOffice, MS-Word κλπ.) θα ήταν ιδιαίτερα χρήσιμο. Αν επιλέξετε χειρόγραφες απαντήσεις, το κείμενο πρέπει να είναι *ευανάγνωστο*.