

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

2ο Σετ Ασκήσεων  
Ημερ. Ανακοίνωσης: 30/10  
Ημερ. Υποβολής: 06/11 – Ώρα 15:00 (στην τάξη)

Δώστε σύντομες, περιεκτικές, και δακτυλογραφημένες απαντήσεις σε όλα στα παρακάτω ερωτήματα:

- Πρόβλημα 1 (10 πόντοι):

Περιγράψτε τι ακριβώς συμβαίνει όταν ένα συμβολομεταφρασμένο πρόγραμμα χρήστη πρέπει να εκτελέσει την παρακάτω εντολή της C Library:

```
fprintf(myfile, "This is a test.%s\n", buffer);
```

Τι ακριβώς συμβαίνει μεταξύ προγράμματος υπό εκτέλεση, C Library και πυρήνα; Διαγραμματικά δώστε την ακολουθία των ενεργειών που συμβαίνουν για να ολοκληρωθεί η εκτέλεση της παραπάνω εντολής.

- Πρόβλημα 2 (20 πόντοι):

Γράψτε ένα πρόγραμμα C που να είναι σε μήκος λιγότερο από 15 γραμμές (πηγαίο κώδικα) και να τυπώνει τις αρχικές διευθύνσεις των περιοχών μνήμης ενός εκτελέσιμου που έχει υιοθετήσει την μορφή ELF-format binary file όπως αυτή έχει παρουσιαστεί στην τάξη. Το πρόγραμμα σας θα πρέπει να δίνει τις αρχικές διευθύνσεις των `text segment` (κώδικα), `initialized data`, `uninitialized data`, `stack` και `heap`.

- Πρόβλημα 3 (10 πόντοι):

Το παρακάτω πρόγραμμα έχει μάλλον μη αναμενόμενο αποτέλεσμα εκτέλεσης:

```
#include <stdio.h>

void AllocateMyArray(int n) {
    int mydata[n];
    printf("Just allocated an array of data with %d elements\n",
           (int)sizeof(mydata));
    printf("Is this for real? \n");
}

int main(){
    AllocateMyArray(50);
    AllocateMyArray(250);
    AllocateMyArray(450);
    printf("WHAT ??? \n");
}
```

Τι είναι ακριβώς αυτό που είναι μη αναμενόμενο; Γιατί συμβαίνει αυτό που βλέπετε όταν συμβολομεταφράζεται το παραπάνω πρόγραμμα με τον `gcc` της διανομής που χρησιμοποιείτε;

- Πρόβλημα 4 (20 πόντοι):

Εξηγήστε τι ακριβώς συμβαίνει και γιατί με το παρακάτω πρόγραμμα που ταξινομεί 5 ακεραίους.

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

#define NUMBER 5

int mycompare(const void * n1, const void * n2) {
    return ( *(int *)n1 - *(int *)n2 );
}

int main(){
    int mynumbers[NUMBER]={123,45,2,456,-34};
    int i;

    qsort(mynumbers, NUMBER, sizeof(int), mycompare);

    for (i=0;i<NUMBER;i++) printf("%d: %d\t",i,mynumbers[i]);
    printf("\n");
    return 0;
}
```

Δώστε πλήρη εξήγηση για την χρήση της έκφρασης στην εντολή `return`.

- Πρόβλημα 5 (20 πόντοι):

Μπορείτε να γράψετε ένα παρόμοιο με το παραπάνω πρόγραμμα που να μπορεί τώρα να ταξινομεί με αλφαι-  
νουμερικό τρόπο ένα σύνολο από 5 strings π.χ. {"Emily", "Margarita", "Alexandra", "Evita", "Katerina"}

Η ταξινόμηση θα πρέπει να γίνει πάντα με την βοήθεια της ρουτίνας βιβλιοθήκης `qsort()`.

- Πρόβλημα 6 (10 πόντοι): Από το Βιβλίο Σιλμπερσφατζ 9η ηλεκτρονική Έκδοση κάντε την Άσκηση 3.14 (σελίδα 152).

- Πρόβλημα 7 (10 πόντοι): Γράψτε ένα πρόγραμμα που να μπορεί να δημιουργεί συνέχεια ταυτόχρονες νέες διεργασίες. Το πρόγραμμα μας θα πρέπει να τυπώνει το μέγιστο αριθμό ταυτόχρονων διεργασιών που με επιτυχία μπορούν να δημιουργήσουν οι κλήσεις `fork()` του προγραμματός σας.

Οι συνήθειες 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 ΠΡΕΠΕΙ ΝΑ είναι γραμμένες σε ένα text formatter (δηλ.  $\LaTeX$ , LibreOffice, MS-Word κλπ.) Χειρόγραφες απαντήσεις ΔΕΝ είναι επιλογή.