

Ασκήσεις 7ης διάλεξης – Pointers

Για την κάθε μία από τις παρακάτω εργασίες:

(α) χρησιμοποιήστε ένα ξεχωριστό project – το κατάλληλο smProject που δίνεται,

(β) βάλτε σχόλια στον κώδικα που εξηγούν τα βήματα της επίλυσης και

(γ) δημιουργήστε μέσα στην smMain κώδικα που θα επιδεικνύει την καλή λειτουργία των συναρτήσεων.

Σημείωση: Στον τελικό κώδικα μην χρησιμοποιήσετε τις `printf` μέσα στις ζητούμενες συναρτήσεις.

Συμβουλές: Διαβάστε προσεκτικά δύο ή τρεις φορές την εκφώνηση. Επιλέξτε περιγραφικά και αυτοεξηγούμενα ονόματα μεταβλητών. Χρησιμοποιήστε καλή στοίχιση. Κατά τη συγγραφή του προγράμματος προσθέστε όσες `printf` χρειαστείτε για να βλέπετε τις τιμές των μεταβλητών ώστε να εντοπίζετε ευκολότερα τυχόν λάθη, πριν παραδώσετε όμως, αφαιρέστε τις “βοηθητικές” και κρατήστε μόνο όσες είναι απαραίτητες.

Σημείωση: Στον τελικό κώδικα μην χρησιμοποιήσετε την `printf` μέσα στις ζητούμενες συναρτήσεις.

Συμβουλές: Διαβάστε προσεκτικά την εκφώνηση. Επιλέξτε περιγραφικά ονόματα μεταβλητών. Χρησιμοποιήστε καλή στοίχιση (formatting) του κώδικα.

ΠΡΟΣΟΧΗ! Μην ξεχάσετε να κατεβάσετε και να χρησιμοποιήσετε τα αντίστοιχα smProject για την κάθε άσκηση!

ΠΡΟΣΕΞΤΕ ΟΠΩΣΔΗΠΟΤΕ ΤΑ ΠΑΡΑΚΑΤΩ

Ο τρόπος με τον οποίο πρέπει να υποβάλλετε ερωτήσεις περιγράφεται εδώ:

<https://qna.c-programming.allos.gr/doku.php?id=qna:technical:questions>

Ο τρόπος με τον οποίο πρέπει να υποβάλλετε τον κώδικα των εργασιών στο σύστημα υποβολής περιγράφεται εδώ:

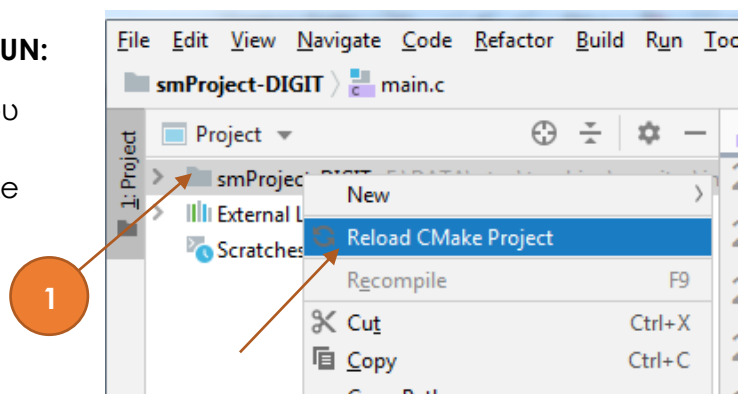
<https://qna.c-programming.allos.gr/doku.php?id=qna:lesson:projects:how-to-submit>

Οι έτοιμες συναρτήσεις της βιβλιοθήκης smLib που διατίθενται για χρήση στα smProjects, περιγράφονται (εκτός από τις διαφάνειες) και στο site ερωταποκρίσεων εδώ:

<https://qna.c-programming.allos.gr/doku.php?id=qna:misc:sm-library>

ΕΑΝ ΔΕΝ ΕΜΦΑΝΙΖΟΝΤΑΙ ΕΝΕΡΓΑ ΤΑ BUILD / RUN:

1. Κάνω **δεξί κλικ** πάνω στο όνομα του project και εμφανίζεται το μενού
2. Κάνω **απλό κλικ** στο Reload CMake Project, δηλαδή τη 2^η επιλογή



Εργασία 7α – Διάταξη ορισμάτων

Βαθμός δυσκολίας: **1/3**

Όνομα smProject: **smProject-IN-ORDER**

Περιγραφή

Ζητείται να γράψετε την παρακάτω συνάρτηση

```
bool inOrder(double *a1, double *a2, double *a3)
```

η οποία δέχεται τρεις pointers σε double και αναδιατάσσει τις τιμές τους ώστε $a1 < a2 < a3$ και επιστρέφει αληθές. Εάν δοθεί ως NULL έστω ένας από τους pointers, τότε θα πρέπει να μην κάνει κάποια αλλαγή και να επιστρέφει ψευδές.

Εργασία 7β – Ελάχιστη τιμή λίστας

Βαθμός δυσκολίας: **1/3**

Όνομα smProject: **smProject-MIN-OF-LIST**

Περιγραφή

Δίνεται η δομή (struct)

```
typedef struct _node {
    struct _node *nextNode;
    double data;
} Node;
```

όπου κάθε μία τέτοια δομή (ας την πούμε κόμβο) περιέχει έναν pointer για μια επόμενη όμοια δομή (επόμενο κόμβο), δημιουργώντας έτσι μια αλυσίδα κόμβων. Ο τελευταίος κόμβος της αλυσίδας έχει τιμή του nextNode ίση με NULL. Καλείστε να γράψετε τη συνάρτηση:

```
double minOfList(Node *firstNode)
```

η οποία επιστρέφει **την ελάχιστη** τιμή του πεδίου data που θα βρεθεί στην αλυσίδα που ξεκινά με τον κόμβο firstNode. Εάν η αλυσίδα δεν έχει κανένα κόμβο, δηλαδή η firstNode είναι NULL, τότε θα πρέπει να επιστρέφει την τιμή $+10^{30}$

Εργασία 7γ – Γραφή Τίτλου

Βαθμός δυσκολίας: **1/3**

Όνομα smProject: **smProject-TITLE-CASE**

Καλείστε να γράψετε τη συνάρτηση

```
bool titleCase(char *text)
```

που να μετατρέπει μια πρόταση ώστε να έχει το 1^ο γράμμα κάθε λέξης κεφαλαίο και όλα τα υπόλοιπα πεζά. Εάν σε ένα κείμενο δεν κάνει καμία αλλαγή επιστρέφει false, αλλιώς επιστρέφει true, αφού ολοκληρώσει βέβαια τις αλλαγές. Ψευδές επίσης επιστρέφεται και όταν ως κείμενο δοθεί η τιμή NULL.

Ένα παράδειγμα μιας πρότασης:

```
This is a SIMPLE test, a tEst for the Longest word OF this Super-text!
```

Από το παραπάνω κείμενο θα πρέπει να προκύπτει το:

```
This Is A Simple Test, A Test For The Longest Word Of This Super-Text!
```

Προσοχή! Η άσκηση αυτή πρέπει να υλοποιηθεί με pointers, χωρίς καθόλου τη χρήση αγκύλων []