



Άσκηση 1 C & Dev-C++

Σκοπός της πρώτης αυτής εργαστηριακής άσκησης είναι να σας εξοικειώσει με το περιβάλλον ανάπτυξης Dev-C++, με το οποίο θα δουλέψετε σ' αυτό το εργαστήριο, αλλά και με μερικά πρώτα βασικά στοιχεία προγραμματισμού σε C.

1. Δημιουργήστε στη επιφάνεια εργασίας ένα φάκελο με όνομα *ΑριθμόΜητρώου/MyCProgs/Lab1*.
2. Ξεκινήστε το περιβάλλον C επιλέγοντας Start -> All Programs -> Bloodshed Dev-C++ → Dev-C++. Πρόκειται, όπως βλέπετε, για ένα παραθυρικό περιβάλλον.
3. Το παράθυρο που βλέπετε μοιάζει πολύ με τα παράθυρα στα οποία έχετε ήδη συνηθίσει. Οι επιλογές των menus είναι παρόμοιες, το ίδιο και οι συνδυασμοί των πλήκτρων για την εκτέλεση των βασικών λειτουργιών.
4. Ανοίξτε ένα νέο *αρχείο πηγαίου κώδικα (Source file)* επιλέγοντας File -> New -> Source file. Έχει όνομα Untitled1. Με File -> Save As... επιλέγετε να το αποθηκεύσετε στο φάκελο που έχετε δημιουργήσει, δίνοντάς του το όνομα «prog1.c».
5. Γράψτε στο αρχείο που μόλις ανοίξατε το παρακάτω απλό πρόγραμμα (ακριβώς όπως δίνεται):

```
#include <stdio.h>

main ()

{ /* Note the indentation */
printf("Hello, world!\n");
}
```

Το τμήμα εκείνο του περιβάλλοντος που χρησιμοποιείτε τώρα είναι ο *συντάκτης (editor)*. Αποθηκεύστε ό,τι γράψατε (File -> Save ή κλικ στο κουμπί που δείχνει αποθήκευση σε δισκέττα).

6. Έχετε τώρα γράψει και αποθηκεύσει τον *πηγαίο κώδικα* του προγράμματός σας. Με την επιλογή Execute -> Compile καλείτε το *μεταγλωττιστή (compiler)* για να μετατρέψει το πρόγραμμά σας σε γλώσσα αντιληπτή από τον Η/Υ. Το παράθυρο διαλόγου που εμφανίζεται σας δίνει πληροφορίες για το αποτέλεσμα της μεταγλώττισης (πόσα σφάλματα και πόσες προειδοποιήσεις υπάρχουν). Πατήστε Close.
7. Ζητείστε τώρα να εκτελεστεί («τρέξει» (Run)) το πρόγραμμά σας, με την επιλογή Execute -> Run.

(Αν δεν βλέπετε και πολλά πράγματα από την εκτέλεση. Αυτό οφείλεται στο ότι το παράθυρο εκτέλεσης κλείνει αμέσως μετά το τέλος του προγράμματος και δεν προλαβαίνετε να το δείτε. Για να λυθεί αυτό το πρόβλημα, προσθέστε κάτω από την printf τη γραμμή system("pause"); . (Θα χρειαστεί επίσης να προσθέσετε στην αρχή του αρχείου και την οδηγία #include <stdlib.h> πριν κάνετε save.) Παρεμπιπτόντως, καλό είναι, για λόγους καλού στυλ προγραμματισμού (δεν είναι απαραίτητο), η νέα εντολή να ξεκινάει ακριβώς κάτω από την προηγούμενη (στοίχιση (indentation)). Αυτό μπορείτε να το (απο)ενεργοποιείτε με το συνδυασμό πλήκτρων (Shift+Ctrl+U) Shift+Ctrl+I.)

8. Αν και πλέον το πρόγραμμα εκτελείται όπως θα θέλαμε, υπάρχει κι άλλη μια λεπτομέρεια που πρέπει να φροντίσουμε. Η συνάρτηση main ορίστηκε να



επιστρέφει κάτι τύπου int (αυτό, όπως θα δούμε αργότερα, υπονοείται από το ότι δεν γράφουμε τίποτε αριστερά της λέξης main) αλλά δεν έχουμε συμπεριλάβει κάποια εντολή επιστροφής αποτελέσματος. Δεν είναι απαραίτητο (γι' αυτό και το πρόγραμμα λειτούργησε) αλλά για να σεβαστείτε τους τύπους προσθέστε κάτω από τη system("pause") την εντολή (Στο εξής έτσι θα τελειώνετε πάντα τη main) **return(0);** και κάντε save.

9. Ξανακάντε μεταγλώττιση κι εκτελέστε («τρέξτε») το πρόγραμμα. Επαληθεύστε ότι στο φάκελό σας υπάρχει και το εκτελέσιμο πρόγραμμα, prog1.exe. Μπορείτε να το ξανατρέξετε και με διπλό κλικ στο εικονίδιό του.
10. Δοκιμάστε τώρα να εισάγετε λάθη στο πρόγραμμα. Για παράδειγμα, παραλείψτε το ; απ' το τέλος της printf και το τέλος της return και ξανακάντε μεταγλώττιση. Δείτε πώς ο μεταγλωττιστής αντιδρά στα λάθη που κάνατε. Μάλιστα, σας δείχνει σε ποιο τμήμα του πηγαίου κώδικα αντιστοιχεί το μήνυμα λάθους που βλέπετε. Επίσης, με διπλό κλικ στο μήνυμα λάθους μεταφέρεστε στο αντίστοιχο τμήμα του κώδικα. Διορθώστε τα λάθη και ξανακάντε τη μεταγλώττιση και το τρέξιμο του προγράμματος. Δοκιμάστε τώρα να σβήσετε το f από την printf. Τι μήνυμα λάθους παίρνετε; Διορθώστε το.
11. Εισάγετε σχόλιο μέσα στο σχόλιο, ως εξής /* Note the /* following */ indentation */. Μεταγλωττίστε τον κώδικά σας. Τα μηνύματα σφάλματος του μεταγλωττιστή οφείλονται στο ότι η πρότυπη (ANSI) C απαγορεύει το «φώλιασμα» (nesting) των σχολίων. Διορθώστε αυτό το λάθος.
12. Μάθετε τη χρήση του χαρακτήρα αλλαγής γραμμής (newline) \n (Οι χαρακτήρες της μορφής \c λέγονται ακολουθίες διαφυγής (escape sequences)). Προσθέστε στη συμβολοσειρά που δίνεται στην printf άλλη μια πρόταση, π.χ. αλλάξτε την σε printf("Hello, world!\nHello again."); και τρέξτε το νέο πρόγραμμα. Τι κάνει; Εισάγετε ένα \n αμέσως πριν το again. Τι δίνει τώρα; Σβήστε τώρα το backslash, δηλαδή το \, από το ένα \n και ξανατρέξτε το. Δείτε τη διαφορά. Σβήστε και το \n και ξαναδοκιμάστε. Δοκιμάστε να γράψετε την printf ως

```
printf("Hello, world!
```

```
Hello again.");
```

και μεταγλωττίστε. Είδατε γιατί χρειάζεται ο χαρακτήρας \n; (Προσέξτε, με την ευκαιρία, πώς ένα σφάλμα μπορεί να δώσει στο μεταγλωττιστή την εντύπωση ότι υπάρχουν περισσότερα.)

13. Δοκιμάστε τώρα, γράφοντας περισσότερες της μιας printf, να τυπώσετε στην οθόνη τις παραπάνω προτάσεις, τη μια δίπλα στην άλλη και κατόπιν τη μια κάτω απ' την άλλη.
14. Γράψτε, σε άλλο αρχείο, με όνομα prog2.c, πρόγραμμα C που να εμφανίζει στην οθόνη το εξής μήνυμα:

```
Hello, my name is
```

```
Το ονοματεπώνυμό σας (με Λατινικούς χαρακτήρες)
```

```
and I am learning C programming.
```

Χρησιμοποιήστε δύο printf.

15. Πειραματιστείτε αλλάζοντας διάφορα τμήματα στον κώδικά σας, εισάγοντας λάθη, εισάγοντας (κι άλλα) σχόλια, γράφοντας τις εντολές σε διαφορετικά σημεία ή διαφορετική σειρά και δείτε τα αποτελέσματα.
16. Δείτε πώς μπορείτε να βρείτε βοήθεια για τις λειτουργίες του Dev-C++ (καθώς και μια σύντομη επισκόπηση του προγραμματισμού σε C) με την επιλογή Help -> Help on Dev-C++ ή με κλικ στο κουμπί ?.