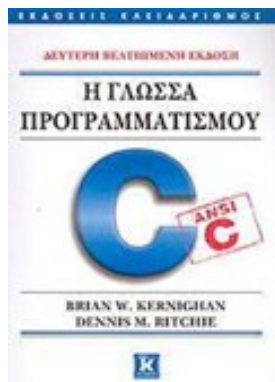


Προγραμματισμός Υπολογιστών & Υπολογιστική Φυσική

Θεόδωρος Σαμαράς

theosama@auth.gr

<http://users.auth.gr/voyatzis/CProg>



Η Γλώσσα Προγραμματισμού C
Brian Kernighan & Dennis Ritchie
2^η έκδοση/2008
ISBN: 978-960-461-132-4
Εκδόσεις Κλειδάριθμος ΕΠΕ

www.c4all.gr

C Από τη θεωρία στην Εφαρμογή
Δρ. Γ. Σ. Τσελίκης | Δρ. Ν. Δ. Τσελίκης

Αρχική σελίδα
Περιοδική
Περιεχόμενα βιβλίου
Κάποιες Διαφορές
Ενδεικτικές Ασκήσεις
Επικοινωνία

ρία στην Εφαρμογή

Τίτλος: C: από τη θεωρία στην Εφαρμογή (2^η Έκδοση, 2012)
Συγγραφείς: Γ. Σ. Τσελίκης - Ν. Δ. Τσελίκης
ISBN: 978-960-461-132-4
Σελίδες: 62
Διαστάσεις: 17x21
Διεύση: Αλυσία Εθνικωνωνιδων Πανεωτηριου

Μια από τις πιο βασικές ασκήσεις που περιλαμβάνονται στο βιβλίο, 2ης έκδοσης του βιβλίου εδώ ↓

Αν θέλετε, η κλίση, που περιλαμβάνει τη 1^η, 2^η έκδοση, του βιβλίου βρίσκεται εδώ ↓

Οδηγίες:

- 1) Αποσυνδέστε το 1^ο και 2^ο αρχείο στον υπολογιστή σας.
- 2) Αποσυνδέοντας δημιουργείται ο φάκελος "source_code_vXX" ("XX": 1 για την 1^η έκδοση και 2 για τη 2^η, αντίστοιχα), ο οποίος περιέχει τους υποφάκελους "Section_XX" ("XX": ο αριθμός του αντίστοιχου κεφαλαίου του βιβλίου).
- 3) Κάθε υποφάκελος "Section_XX" περιέχει τον ηχητικό κώδικα (αρχείο της μορφής XX_YY.c) με το κείμενο της κάθε άσκησης "YY" του αντίστοιχου κεφαλαίου "XX" του βιβλίου.

Για παράδειγμα, το αρχείο 6_B8.c αποτελεί τον ηχητικό κώδικα με τη λύση της τρίτης άσκησης του 6^{ου} κεφαλαίου του βιβλίου. Το αρχείο αυτό θα βρεθεί μέσα στον υποφάκελο "Section_06", στον οποίο θα περιέχονται τα αρχεία με τον ηχητικό κώδικα όλων των ηχητικών ασκήσεων του 6^{ου} κεφαλαίου του βιβλίου.

Στόχοι Μαθήματος

- Η ανάπτυξη ικανοτήτων και η απόκτηση δεξιοτήτων για τη διαχείριση της διαδικασίας επίλυσης προβλημάτων με προγραμματισμό υπολογιστών.
 - **Κύκλος ζωής προγράμματος** : ανάλυση προβλήματος, σχεδίαση λύσης, κωδικοποίηση της λύσης σε πρόγραμμα, έλεγχος και διόρθωση λαθών, τεκμηρίωση και συντήρηση, απόσυρση του προγράμματος.
- Η θεμελίωση αλγοριθμικής σκέψης στην επίλυση προβλημάτων.
- Η εμπέδωση βασικών αρχών προγραμματισμού με έμφαση στον δομημένο προγραμματισμό.
- Η εκμάθηση της γλώσσας προγραμματισμού C.

ΒΑΣΙΚΕΣ ΕΝΝΟΙΕΣ ΣΤΗΝ ΑΝΑΠΤΥΞΗ ΠΡΟΓΡΑΜΜΑΤΩΝ

Βασικές έννοιες

- Ο υπολογιστής είναι μια ηλεκτρονική συσκευή που έχει την δυνατότητα να εκτελεί εντολές με πολύ μεγαλύτερη ταχύτητα και ακρίβεια από ότι ένας άνθρωπος
 - Λειτουργίες εισόδου/εξόδου δεδομένων
 - Αριθμητικές λειτουργίες
 - Κίνηση πληροφοριών μέσα στη CPU
 - Λογικές ή συγκριτικές λειτουργίες

Βασικές έννοιες

- Πρόβλημα (Problem)
 - Το υπολογιστικό πρόβλημα που καλούμαστε να επιλύσουμε, π.χ., ένας αριθμητικός υπολογισμός, η διαχείριση μιας βάσης δεδομένων, ο έλεγχος ενός ρομπότ, μια ιατρική διάγνωση, η πρόγνωση του καιρού.
- Αλγόριθμος (Algorithm)
 - Μια σαφώς καθορισμένη διαδικασία αποτελούμενη από ένα σύνολο εκτελέσιμων βημάτων που θα τερματίσει με ένα αποτέλεσμα μετά από πεπερασμένο πλήθος βημάτων που θα εκτελεσθούν.
- Πρόγραμμα (Program)
 - Μια κωδικοποίηση του αλγόριθμου σε μια γλώσσα προγραμματισμού
- Διεργασία (Process)
 - Το πρόγραμμα που εκτελείται σε έναν υπολογιστή

Δομημένος προγραμματισμός

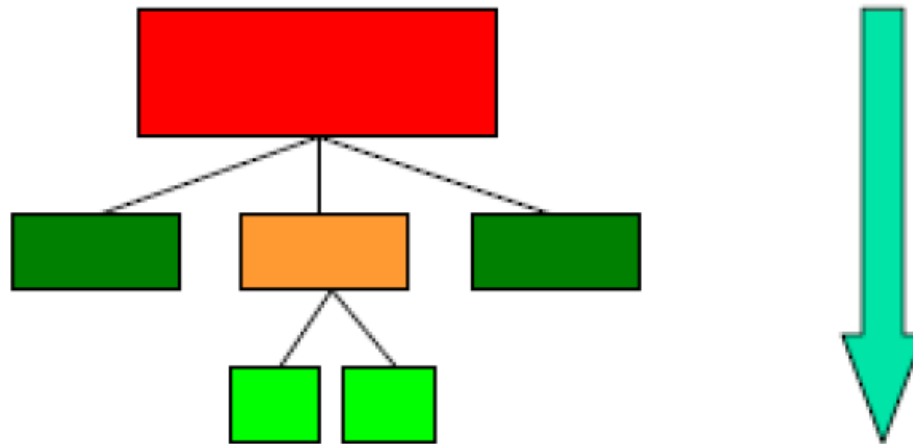
- Ανάλυση του προγράμματος που επιλύει ένα πρόβλημα σε τμήματα (υποπρογράμματα, modules) τα οποία συνεργάζονται για να δώσουν το τελικό αποτέλεσμα, αλλά ταυτόχρονα δημιουργούνται (και ελέγχονται) ανεξάρτητα το ένα από το άλλο.

Δομημένος προγραμματισμός

- Πλεονεκτήματα:
 - Ευκολότερη κατανόηση της λειτουργίας του προγράμματος από τρίτους
 - Ευκολότερος έλεγχος της ορθότητας του προγράμματος
 - Ευκολότερη αποσφαλμάτωση προγράμματος (debugging)
 - Δυνατότητα ανάπτυξης υποπρογραμμάτων ανεξάρτητα, από διαφορετικές ομάδες
 - Δυνατότητα επαναχρησιμοποίησης υποπρογραμμάτων
 - Ευκολότερη συντήρηση

Top-down σχεδιασμός λύσεων

- **Divide-and-conquer:** Διαίρεση μεγάλου (και δύσκολου) προβλήματος σε περισσότερα μικρά (και πιο εύκολα) προβλήματα



Bottom-up σχεδιασμός λύσεων

- Ανάπτυξη των τμημάτων του προγράμματος ξεκινώντας από τα μικρότερα προς τα μεγαλύτερα (που τα περικλείουν).
- Συνηθίζεται περισσότερο στην κωδικοποίηση του αλγορίθμου σε πρόγραμμα.

Πρόβλημα - παράδειγμα

- Έχουμε ένα αρχείο με τα ονόματα των φοιτητών.
- Θέλουμε να βρούμε τους φοιτητές του 1^{ου} έτους.

STUDENTS.DAT

Επίθετο	Όνομα	Ηλεκτρονικό Ταχυδρομείο	Έτος	Τηλέφωνο
Davis	Albert	none	1	668-78-9226
Crane	Amory	none	2	689-48-8430
Schakowsky	Anibal	none	3	652-58-7355
Kirk	Anne	rep.kirk@volcano.com	4	623-87-0203
Weller	Anthony	none	1	780-52-0498
Costello	Barbara	none	1	599-98-7962
Biggert	Barney	none	1	058-86-1065
Hastert	Baron	speaker@volcano.com	1	075-45-4923






Αναπαράσταση αλγόριθμων

- **Ελεύθερο κείμενο** (free text): ανεπεξέργαστος και αδόμητος τρόπος παρουσίασης αλγόριθμου
- **Διαγραμματικές τεχνικές** (diagramming techniques): γραφικός τρόπος παρουσίασης του αλγόριθμου (διάγραμμα ροής)
- **Κωδικοποίηση** (coding): πρόγραμμα γραμμένο είτε σε **ψευδογλώσσα**, είτε σε προγραμματιστικό περιβάλλον που όταν εκτελεσθεί θα δώσει τα ίδια αποτελέσματα με τον αλγόριθμο





Διαγράμματα ροής

- Χρησιμοποιούνται διάφορα γεωμετρικά σχήματα με ορισμένη σημασία
- Τα σχήματα συνδέονται με συνεχείς γραμμές και χρησιμοποιούνται τόξα για να δείξουν την ροή των δεδομένων
- Ένα διάγραμμα ροής δεν είναι ένα συμπληρωμένο πρόγραμμα, αλλά αποτελεί μια γραφική αναπαράσταση των βημάτων ενός αλγόριθμου που μπορεί να χρησιμοποιηθεί ως βοήθεια για τον προγραμματιστή

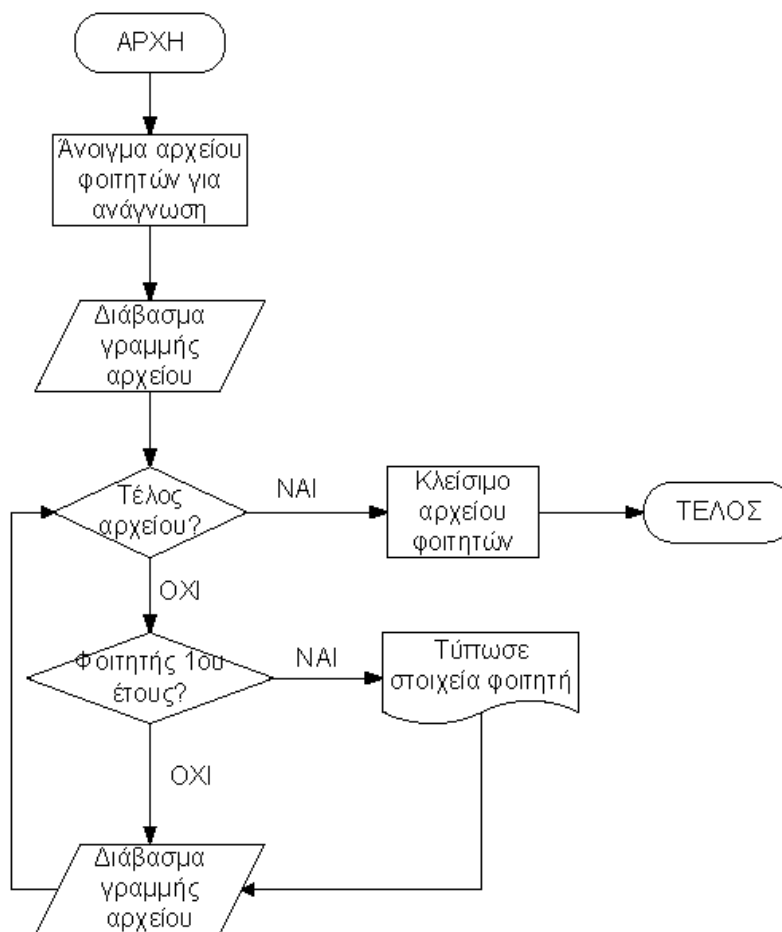
Σύμβολα διαγράμματος ροής

Σύμβολο	Όνομα	Περιγραφή
	Αρχή/Τέλος	Συμβολίζει την αρχή ή το τέλος του διαγράμματος ροής.
	Επεξεργασία	Συμβολίζει την επεξεργασία δεδομένων (μαθηματικές πράξεις, αναθέσεις τιμών, κ.α.)
	Είσοδος/Εξοδος	Συμβολίζει την είσοδο δεδομένων για επεξεργασία και την έξοδο επεξεργασμένων δεδομένων.
	Ροή	Συμβολίζει τη λογική ροή του προγράμματος. Δείχνει το επόμενο σύμβολο η λειτουργία του οποίου θα εκτελεστεί.
	Απόφαση	Συμβολίζει την επεξεργασία λογικών συνθηκών που κτίζονται με τη χρήση σχεσιακών και λογικών τελεστών.

Σύμβολα διαγράμματος ροής

Σύμβολο	Όνομα	Περιγραφή
	Αρχικοποίηση	Συμβολίζει την αρχικοποίηση της μνήμης (μεταβλητών) για επεξεργασία δεδομένων.
	Εκτύπωση	Συμβολίζει την εκτύπωση στην οθόνη ή τον εκτυπωτή.
	Σύνδεση στην ίδια σελίδα	Συμβολίζει μια σύνδεση στο διάγραμμα ροής της ίδιας σελίδας για να αποφευχθεί συνωστισμός συνδέσεων.
	Σύνδεση σε διαφορετική σελίδα	Συμβολίζει μια σύνδεση στο διάγραμμα ροής άλλης σελίδας για να αποφευχθεί συνωστισμός συνδέσεων και υπερφόρτωση συμβόλων.

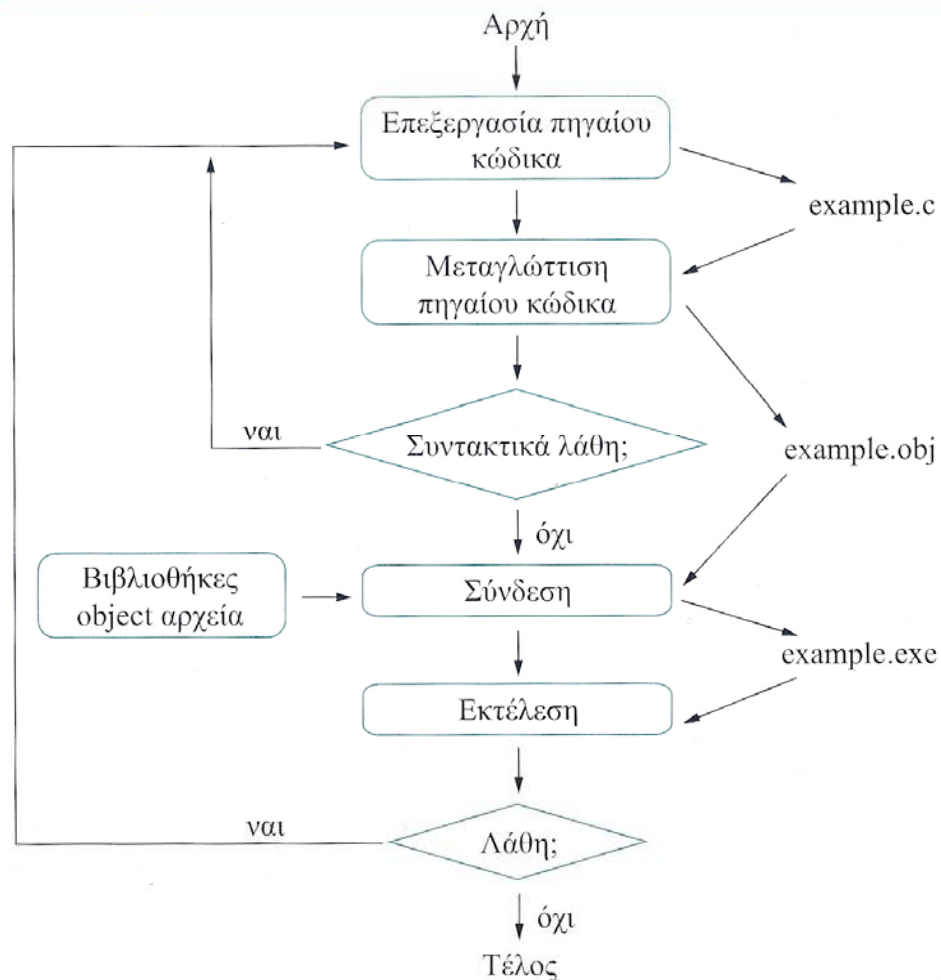
Διάγραμμα ροής για το παράδειγμα



Κώδικας σε C

```
#include <stdio.h>
int main()
{
    FILE *fp;
    char line[100], *c;
    /* open the file */
    fp = fopen("STUDENT.DAT", "r");
    if (fp == NULL) {
        printf("Unable to open this file.\nAborting ... \n");
        exit(-1);
    }
    /* read a line */
    c = fgets(line, 100, fp);
    while (c != NULL) /* Check EOF */
    {
        if (getyear(line) == 1) /* getyear function returns the year */
            printf("%s", line);
        c = fgets(line, 100, fp); /* read the next line */
    }
    fclose(fp);
    return 0;
}
```

Διάγραμμα ροής για μετάφραση – εκτέλεση ενός C προγράμματος



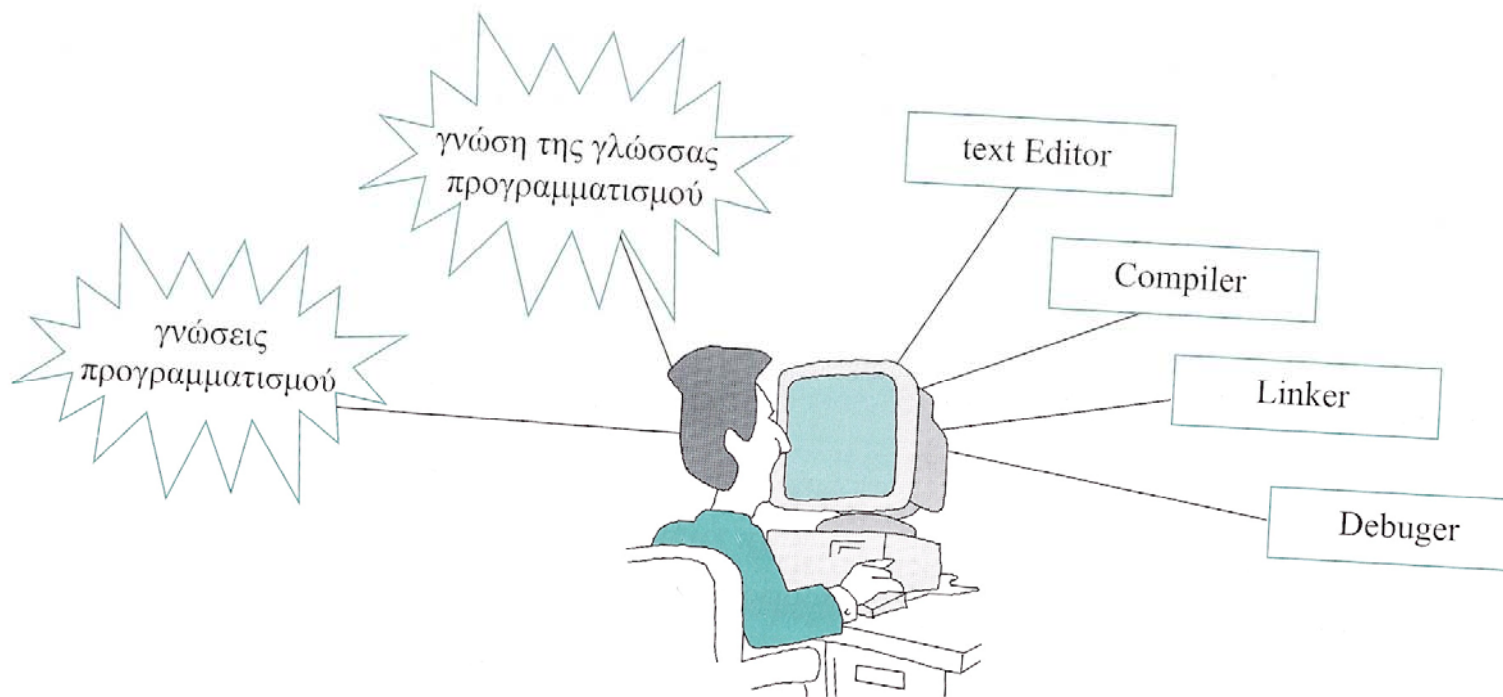
Έλεγχος αποτελεσμάτων

Davis	Albert	none	1	668-78-9226
Weller	Anthony	none	1	780-52-0498
Costello	Barbara	none	1	599-98-7962
Biggert	Barney	none	1	058-86-1065
Hastert	Baron	speaker@volcano.com	1	075-45-4923

- Σε περίπτωση που το αποτέλεσμα δεν είναι το αναμενόμενο θα πρέπει να εντοπισθούν τα **λογικά λάθη** του προγράμματος που συνδέονται με το σχεδιασμό της λύσης
- Το πρόγραμμα θα πρέπει να ελεγχθεί με περισσότερα δεδομένα (εμπειρική απόδειξη ορθότητας έναντι αναλυτικής μεθόδου με μαθηματική απόδειξη).

ΓΛΩΣΣΕΣ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ

Κωδικοποίηση αλγόριθμου



Γλώσσες προγραμματισμού

- Γλώσσες μηχανής
 - Άμεση επικοινωνία με το υλικό του υπολογιστή
- Συμβολικές γλώσσες
 - Χρήση συμβολικών ονομάτων για τις βασικές εντολές
- Υψηλού επιπέδου
 - Προσανατολισμένες στο πρόβλημα και όχι στον υπολογιστή

Γλώσσες μηχανής

- Η μόνη γλώσσα που «καταλαβαίνει» άμεσα ο υπολογιστής
- Κάθε υπολογιστής έχει τη δική του διαφορετική γλώσσα μηχανής
- Κάθε κωδικός της γλώσσας μηχανής συμβολίζει μία συγκεκριμένη λειτουργία
 - π.χ. Πολλαπλασιασμός = 1001
- Ο προγραμματισμός σε γλώσσα μηχανής είναι εξαιρετικά επίπονη διαδικασία
- Ξένη προς τον τρόπο που σκέφτεται ο άνθρωπος
- Μεγάλη ταχύτητα εκτέλεσης των προγραμμάτων

Συμβολικές γλώσσες

- Για την καλύτερη απομνημόνευση των κωδικών μηχανής χρησιμοποιούνται μνημονικοί κωδικοί
 - π.χ. Πολλαπλασιασμός = MULT = 1001
- Η μετάφραση από τα συμβολικά ονόματα στους αριθμητικούς κωδικούς γίνεται μέσω του συμβολομεταφραστή (**assembler**)

Γλώσσες υψηλού επιπέδου

- Δεν εξαρτώνται από την μηχανή αλλά από το πρόβλημα
- Επιτρέπουν στον προγραμματιστή να γράψει εντολές χρησιμοποιώντας κωδικές λέξεις και συμβατικούς μαθηματικούς συμβολισμούς
 - π.χ. $D = A * B + C$
- Επιτρέπουν στον προγραμματιστή να εκφράσει σε υψηλό επίπεδο δομές ελέγχου ροής: ακολουθιακή, επιλογής και επανάληψης
- Cobol, Fortran, Algol, Ada, Pascal, C, C++, Java, C#, Visual Basic, SQL, Prolog, PHP, Perl, Python

Μεταγλώττιση (compilation)

- Ο υπολογιστής δεν «καταλαβαίνει» άμεσα μια γλώσσα υψηλού επιπέδου
- Η μετάφραση από μία γλώσσα υψηλού επιπέδου στη γλώσσα μηχανής γίνεται μέσω των μεταγλωττιστών (compilers)
 - Οδηγίες του προγραμματιστή: αποθηκευμένες στο **πηγαίο πρόγραμμα** (source program)
 - Αντίστοιχες οδηγίες επιπέδου μηχανής: αποθηκευμένες στο **αντικειμενικό ή τελικό πρόγραμμα** (object program)
 - Σε ορισμένα μικρά υπολογιστικά συστήματα (π.χ. PC) το αντικειμενικό πρόγραμμα πρέπει να προετοιμαστεί για εκτέλεση από άλλο ειδικό πρόγραμμα, τον **συνδέτη** (linkage editor)

Ιστορική αναδρομή της C

1969-1973	AT & T Bell Labs, Dennis Ritchie
1978	“The C Programming Language” K & R: Kernighan & Ritchie
1983	Σύσταση ANSI Standardization Committee X3J11
1989-1990	Αποδοχή ANSI/ISO Standard (ANSI C)
1990-1999	Αναθεώρηση του standard υπό εξέλιξη C9X (C99)

Χαρακτηριστικά της C

- Είναι σχετικά μικρή και εύκολη στην εκμάθηση
- Υποστηρίζει top-down και bottom-up σχεδιασμό αλλά και δομημένο προγραμματισμό
- Είναι αποτελεσματική (οδηγεί σε γρήγορα στην εκτέλεση προγράμματα)
- Είναι φορητή (portable) σε πολλά συστήματα
- Ο προγραμματιστής έχει τον πλήρη έλεγχο, αλλά και την ευθύνη των σφαλμάτων (π.χ. διαχείριση δεικτών και δυναμικής μνήμης)

Χαρακτηριστικά της C

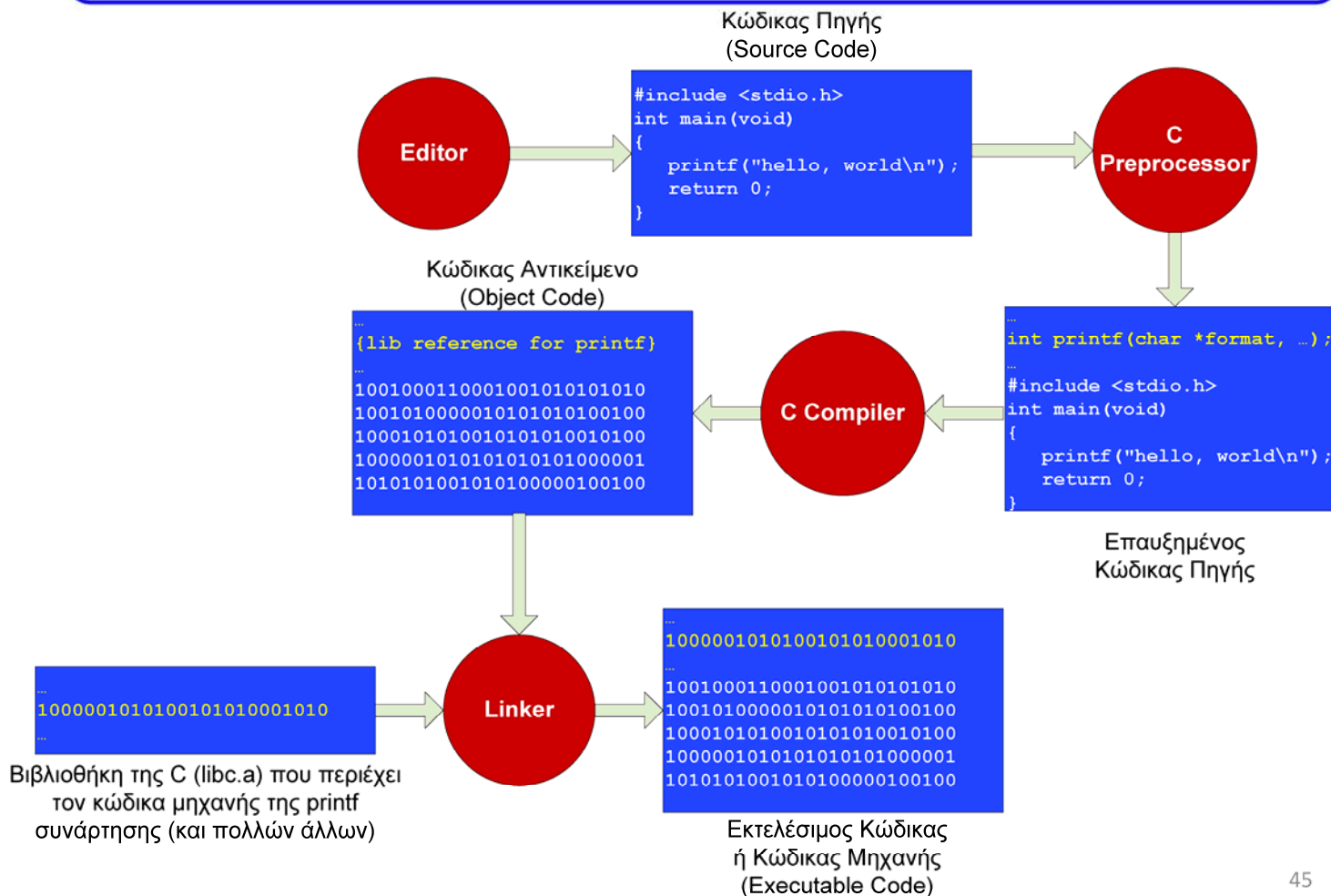
- Υπάρχει μεγάλη βάση εφαρμογών που αναπτύχθηκαν με τη γλώσσα αυτή
- Μπορεί να χρησιμοποιηθεί και ως γλώσσα χαμηλού επιπέδου (άμεση πρόσβαση στους πόρους) και ως γλώσσα υψηλού επιπέδου (διαθέσιμες βιβλιοθήκες)
- Δημιουργήθηκε για την ανάπτυξη του UNIX
- Η γνώση της C είναι καλό εφόδιο για την εκμάθηση άλλων γλωσσών όπως: C++, Java και PHP.

Απλό πρόγραμμα σε C

```
/*  
    Hello world program in C  
*/  
  
#include <stdio.h>  
  
int main()  
{  
    printf("Hello, world!\n");  
  
    getchar(); /* Τρυκ, για να παγώσει το πρόγραμμα πριν  
                τερματίσει, επιτρέποντάς μας να δούμε  
                την έξοδό του. Περιμένει π.χ. return για  
                να ξεπαγώσει. Σε Windows μπορεί να  
                χρησιμοποιηθεί η εντολή system("PAUSE")  
                η οποία όμως πάσχει από φορητότητα. */  
  
    return 0;  
}
```

44

Διαδικασία μεταγλώττισης



Επεξήγηση κώδικα

```
/*  
    Hello world program in C  
*/
```

```
/* Τρυκ, για να παγώσει το πρόγραμμα πριν  
τερματίσει, επιτρέποντάς μας να δούμε  
την έξοδό του. Περιμένει π.χ. return για  
να ξεπαγώσει. Σε Windows μπορεί να  
χρησιμοποιηθεί η εντολή system("PAUSE")  
η οποία όμως πάσχει από φορητότητα. */
```

- Οτιδήποτε περικλείεται ανάμεσα σε `/*` και `*/` υποδηλώνει σχόλιο του προγραμματιστή σχετικά με το πρόγραμμα
- Τα σχόλια δεν μεταγλωττίζονται, αλλά αφαιρούνται κατά την διάρκεια της προεπεξεργασίας (C Preprocessor)

Επεξήγηση κώδικα

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

- Οδηγία προς τον C Preprocessor να συμπεριλάβει τα περιεχόμενα του αρχείου επικεφαλίδας (header file) **stdio.h** το οποίο περιέχει τις δηλώσεις συναρτήσεων της C βιβλιοθήκης για είσοδο/έξοδο στην κονσόλα.
- Στο πρόγραμμα μας τέτοιες συναρτήσεις είναι οι **printf()**, **getchar()**

Επεξήγηση κώδικα

```
int main()
{
  ...
}
```

- Κάθε έγκυρο C πρόγραμμα πρέπει να διαθέτει μια συνάρτηση **main()**.
- Από αυτή την συνάρτηση αρχίζει η εκτέλεση του προγράμματος μετά την μεταγλώττιση.
- Όταν ολοκληρωθεί η εκτέλεση των εντολών της `main()` από τον υπολογιστή, μια ακέραια τιμή (**int**) επιστρέφεται στο περιβάλλον από όπου έγινε η εκκίνηση της εκτέλεσης του προγράμματος και τελικά στον χρήστη.

Επεξήγηση κώδικα

```
printf("Hello, world!\n");
```

- Η συνάρτηση `printf()` μας παρέχεται ως έτοιμη λειτουργία καθώς ορίζεται σε μια από τις βιβλιοθήκες της C (`libc.a`)
- Τυπώνει στην οθόνη τη συμβολοσειρά που λαμβάνει ως όρισμα.
 - Η συμβολοσειρά ορίζεται μεταξύ των διπλών εισαγωγικών " και ".
- Η ακολουθία `\n` σημαίνει newline (νέα γραμμή) και υποδηλώνει πως η επόμενη εκτύπωση χαρακτήρων στην οθόνη θα γίνει σε νέα γραμμή.
- Το ερωτηματικό `;` στο τέλος της πρότασης είναι μέρος της σύνταξης της C για να τερματίσει μια εντολή.

Επεξήγηση κώδικα

```
getchar(); /* Τρυκ, για να παγώσει το πρόγραμμα πριν  
τερματίσει, επιτρέποντάς μας να δούμε  
την έξοδό του. Περιμένει π.χ. return για  
να ξεπαγώσει. Σε Windows μπορεί να  
χρησιμοποιηθεί η εντολή system("PAUSE")  
η οποία όμως πάσχει από φορητότητα. */
```

- Η συνάρτηση **getchar()** διαβάζει ένα χαρακτήρα από την κύρια είσοδο (πληκτρολόγιο) και τον επιστρέφει στο πρόγραμμα ως μια ακέραια τιμή.
- Η επιστρεφόμενη τιμή της `getchar()` που πηγαίνει στο συγκεκριμένο πρόγραμμα;

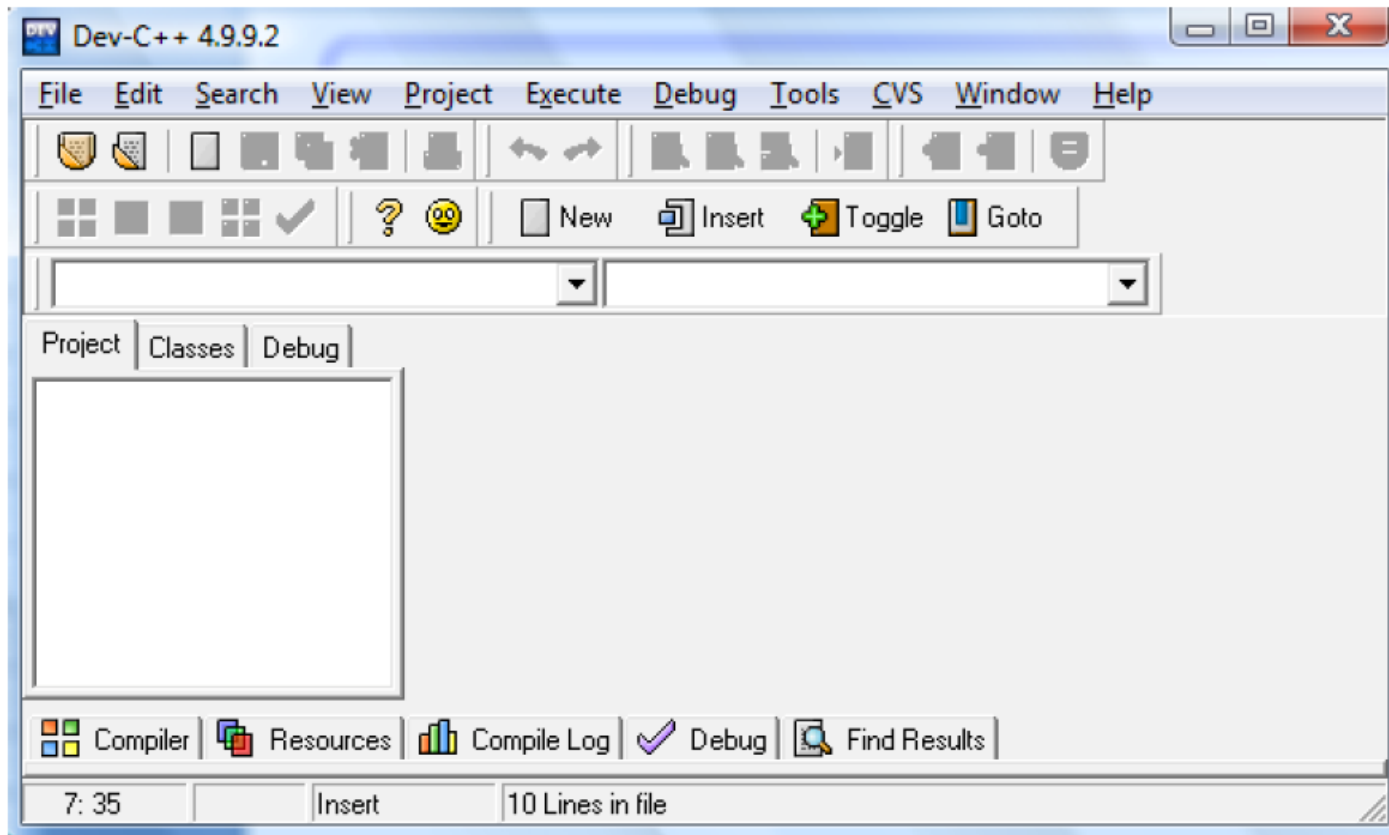
Περιβάλλοντα C προγραμματισμού

- Περιβάλλον Microsoft Visual C++ Express Edition για Windows
- Μεταγλωττιστής GNU gcc για UNIX (ελεύθερο λογισμικό)
- Σε Windows τα παρακάτω περιβάλλοντα υποστηρίζουν τον μεταγλωττιστή gcc
 - Dev-C++
 - Cygwin
 - MinGW

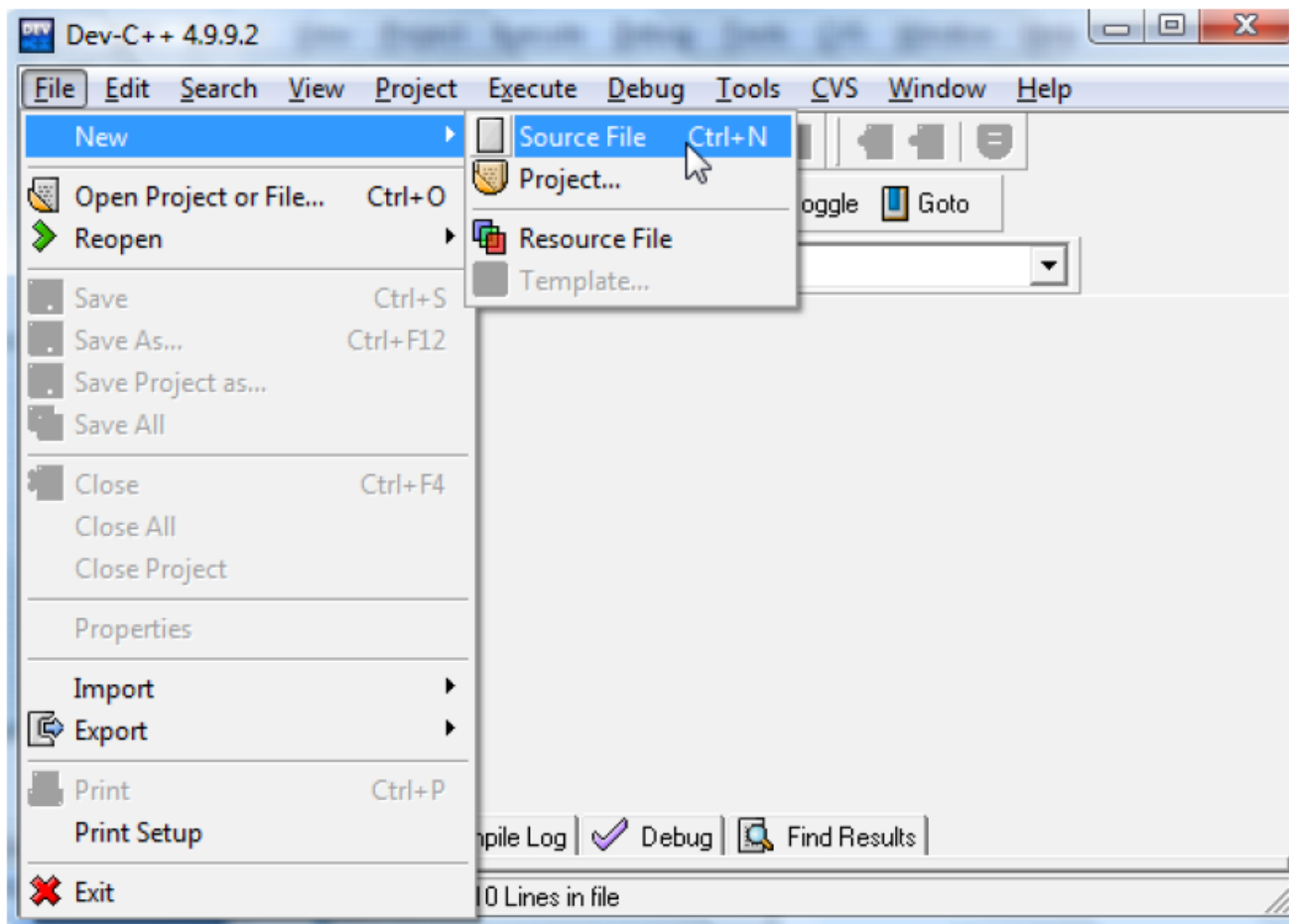
Dev-C++

- Ανοικτό λογισμικό (GNU General Public License)
- **Integrated Development Environment** για C/C++ προγράμματα
- Χρησιμοποιεί την MinGW υλοποίηση του gcc
- Η τελευταία έκδοση μπορεί να βρεθεί στην παρακάτω διεύθυνση:
<http://www.bloodshed.net/dev/devcpp.html>
 - Dev-C++ 5.0 beta 9.2 (4.9.9.2) (9.0 MB) with Mingw/GCC 3.4.2
 - Κατεβάζετε ένα setup αρχείο της μορφής devcpp-4.9.9.2_setup.exe

Το περιβάλλον Dev-C++

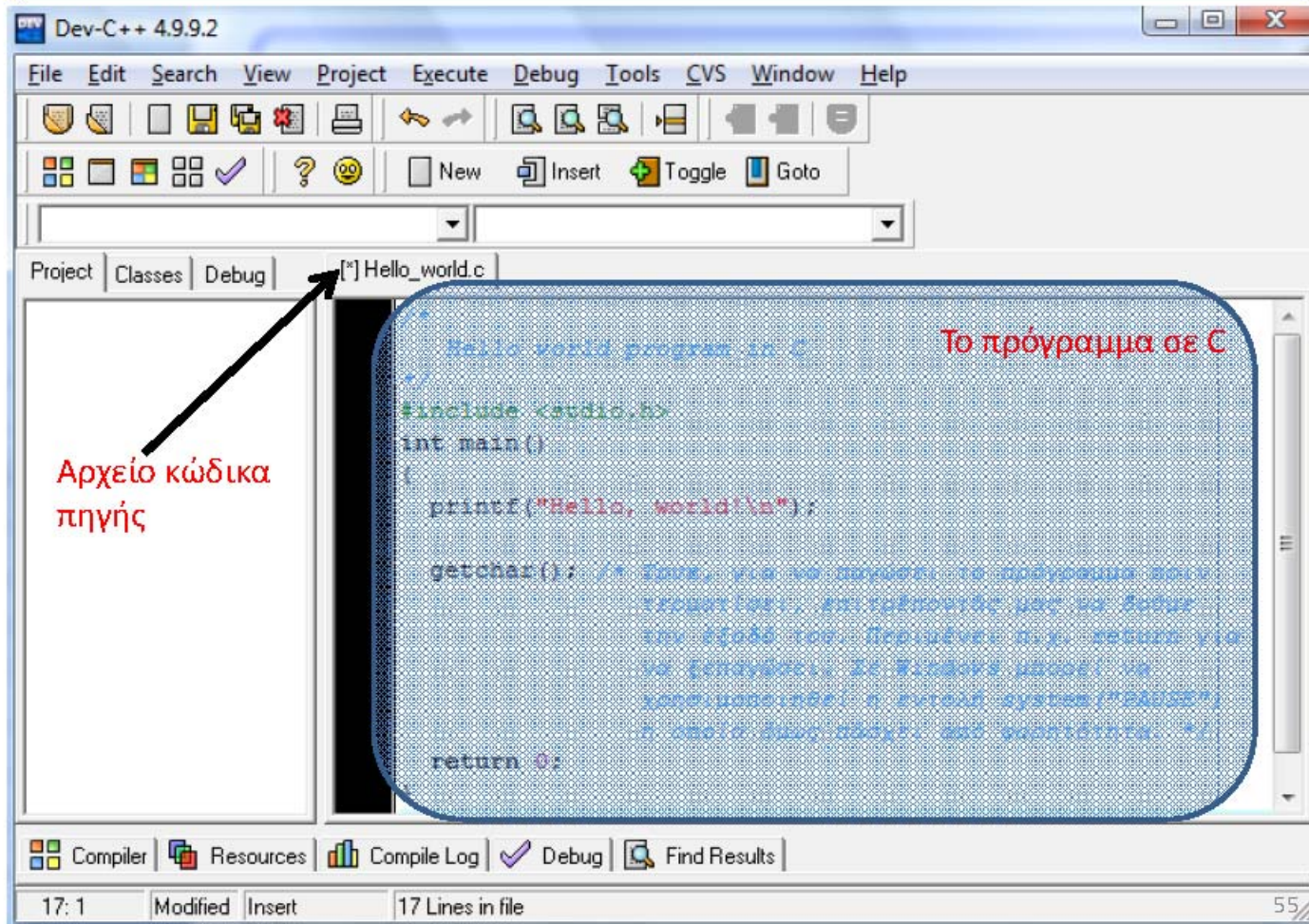


Δημιουργία αρχείου κώδικα πηγής



54

Συγγραφή προγράμματος



The screenshot shows the Dev-C++ 4.9.9.2 IDE. The main window displays a C program titled "Hello_world.c". The code is as follows:

```
/* Hello world program in C */
#include <stdio.h>
int main()
{
    printf("Hello, world!\n");

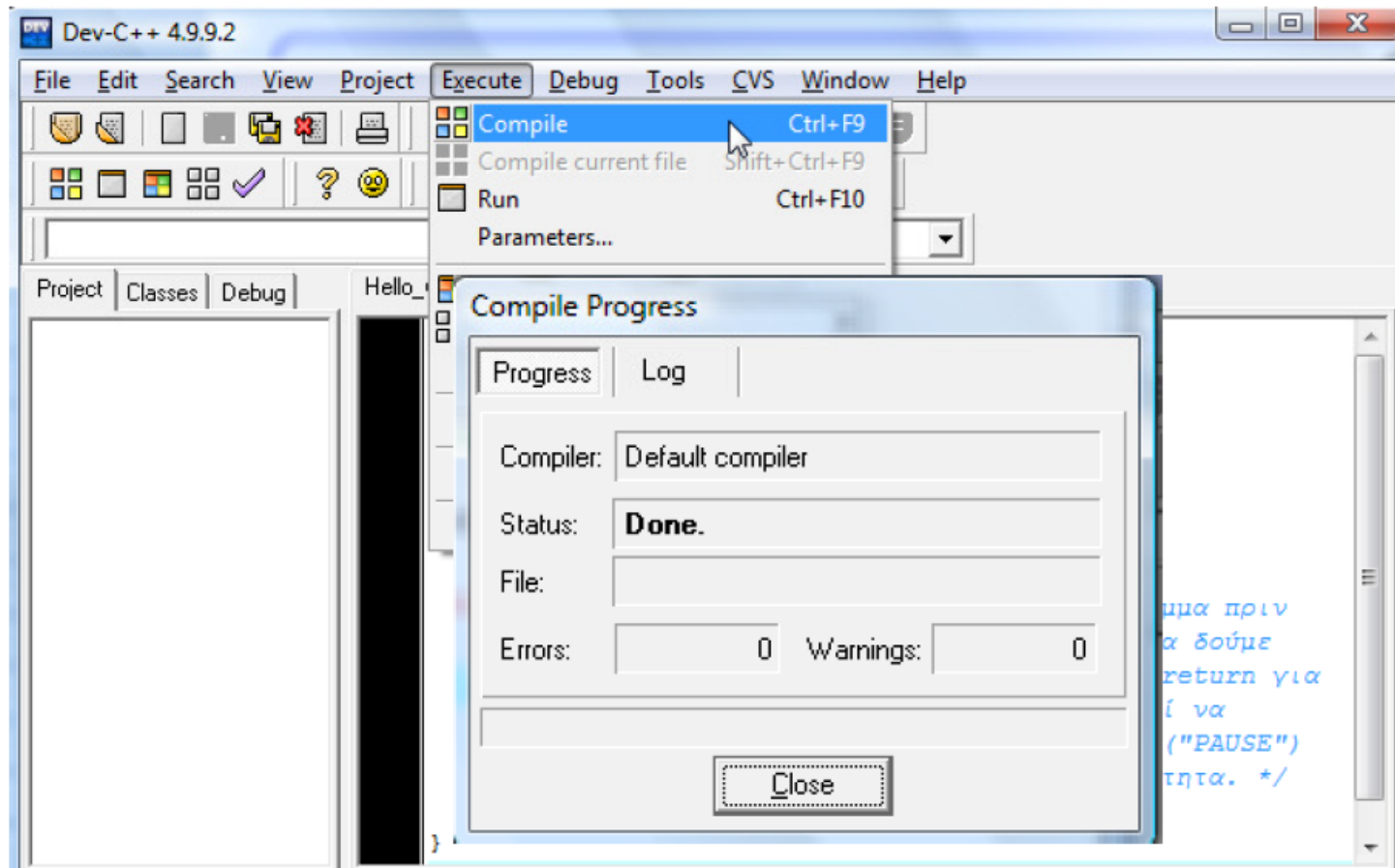
    getch(); /* Take, via the getch() to program some
             * characters. The getch() must be before
             * the printf() to ensure that the output
             * is displayed. In Windows mode, the
             * getch() is not needed. In UNIX mode,
             * the getch() is needed. */
    return 0;
}
```

Annotations on the image:

- A red arrow points from the text "Αρχείο κώδικα πηγής" (Source code file) to the tab labeled "[*] Hello_world.c".
- A blue rounded rectangle highlights the code, with the text "Το πρόγραμμα σε C" (The program in C) written in red to its right.

At the bottom of the IDE, the status bar shows "17: 1 Modified Insert 17 Lines in file" and the page number "55".

Μεταγλώττιση προγράμματος



Εκτέλεση προγράμματος

