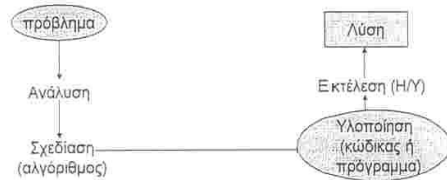


ΕΙΣΑΓΩΓΗ ΣΤΟΝ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟ

ΕΡΓΑΣΤΗΡΙΟ 1

1

Από το Πρόβλημα στο Πρόγραμμα



2

Μερικές βασικές έννοιες

Αλγόριθμος

- Μία υπολογιστική διαδικασία που αποτελείται από μία πεπερασμένη σειρά συστηρώς καθορισμένων υπολογιστικών βημάτων με σκοπό την λύση ενός προβλήματος.
- Κάθε αλγόριθμος δέχεται κάποια δεδομένα σαν είσοδο και παράγει κάποια αποτελέσματα (τη λύση του προβλήματος) σαν έξοδο.
- Τρόποι παρουσίασης ενός αλγόριθμου : Φυσική Γλώσσα, Διάγραμμα Ροής

3

Πρόγραμμα – Γλώσσα Προγραμματισμού

- Πρόγραμμα είναι η έκφραση ενός αλγόριθμου σε μία γλώσσα προγραμματισμού
- Γλώσσα προγραμματισμού είναι μία συστηματική σημειογραφία με την οποία περιγράφουμε με ακρίβεια υπολογιστικές διεργασίες ή διαδικασίες
- Υπολογιστική διαδικασία ή διεργασία είναι ένα σύνολο από υπολογιστικά βήματα που μπορεί να εκτελέσει ένας υπολογιστής προκειμένου να διεκπεραιώσει ένα συγκεκριμένο έργο

4

Παράδειγμα

Πρόβλημα

- Να βρεθεί ο μέσος όρος N πραγματικών αριθμών

Ανάλυση

- Δεδομένα: οι αριθμοί a_1, a_2, \dots, a_N
- Ζητούμενο: ο μέσος όρος

Σχεδίαση-αλγόριθμος

1. Διάβασε τα δεδομένα
2. Υπολόγισε τον μέσο όρο
3. Εκτύπωσε το αποτέλεσμα

5

Συνέχεια παραδείγματος

Σχεδίαση-αλγόριθμος

1. Θέσε την τιμή μηδέν στο Σ
2. Επανάλαβε το παρακάτω N φορές:
διάβασε έναν αριθμό από τα δεδομένα και πρόσθεσέ τον στο άθροισμα Σ
3. Διάρθεσε το άθροισμα Σ με το πλήθος των αριθμών N και καταχώρησε το αποτέλεσμα στο MO
 $MO = \Sigma / N$
4. Εκτύπωσε την τιμή του MO

6

Τέλος παραδείγματος

Υλοποίηση-Πρόγραμμα

```
#include <stdio.h>
main()
{
    float s=0,a,m;
    int i;
    for(i=0;i<N;i++)
    {
        scanf("%d",&a);
        s+=a;
    }
    m=s/N;
    printf("%f",m);
}
```

7

Από την υλοποίηση στην εκτέλεση



8

Πηγαίο πρόγραμμα

- Είναι το αρχικό πρόγραμμα που γράφεται από τον προγραμματιστή (**source code**)
- Για την σύνταξη του απαιτείται απλά ένας κειμενογράφος **Editor** (δεν είναι απαραίτητη στο στάδιο αυτό η ύπαρξη της γλώσσας προγραμματισμού στον υπολογιστή μας)
- Στην περίπτωση της γλώσσας C την οποία χρησιμοποιούμε αποθηκεύεται σαν **αρχείο** με κατάληξη **.c**

9

Μεταγλωττιστής (Compiler)

- Είναι ένα πρόγραμμα το οποίο δέχεται σαν είσοδο ένα πρόγραμμα γραμμένο σε μία γλώσσα προγραμματισμού και παράγει ένα ισοδύναμο πρόγραμμα σε **γλώσσα μηχανής** (δηλαδή σε εντολές σε δυαδική μορφή τις οποίες μπορεί να εκτελέσει ο υπολογιστής)
- Το αποτέλεσμα της μεταγλώττισης είναι ο **αντικειμενικός κώδικας** (αρχείο με κατάληξη **.obj**)
- Στο στάδιο της μεταγλώττισης γίνεται και η **εκσφαλμάτωση**
- Το πρόγραμμα που παράγει ο μεταγλωττιστής **δεν είναι εκτελέσιμο**

10

Εκσφαλμάτωση

Τα λάθη τα οποία μπορεί να υπάρξουν σε ένα πρόγραμμα είναι

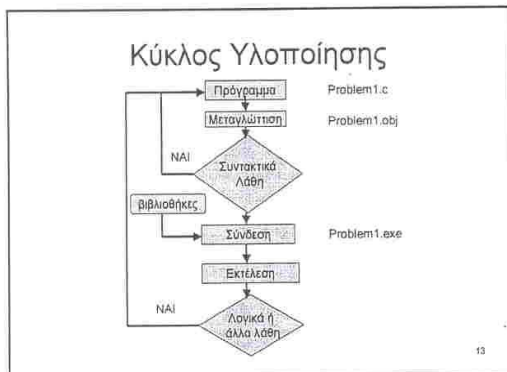
- **Συντακτικά:** τα λάθη αυτά ανιχνεύονται από τον μεταγλωττιστή εμφανίζοντας τα κατάλληλα διαγνωστικά μηνύματα
- **Λογικά:** τα λάθη αυτά ανιχνεύονται από τον προγραμματιστή κατά τη διάρκεια και μετά το τέλος της εκτέλεσης του προγράμματος

11

Συνδέτης (Linker)

- Είναι ένα ειδικό πρόγραμμα το οποίο **συνδέει** το αντικείμενο πρόγραμμα με άλλα τμήματα προγράμματος που βρίσκονται στις βιβλιοθήκες της γλώσσας ή τα έχει γράψει ο προγραμματιστής
- Το αποτέλεσμα είναι το τελικό **εκτελέσιμο πρόγραμμα** (executable)

12



Borland C++ Builder 5

- Το εργαλείο που θα χρησιμοποιήσουμε για να αναπτύξουμε τα προγράμματά μας
- Η γλώσσα προγραμματισμού που θα χρησιμοποιηθεί στο εξάμηνο αυτό είναι η C
- Εκκίνηση του Builder

ΔΗΜΙΟΥΡΓΙΑ ΠΡΟΓΡΑΜΜΑΤΟΣ

Για να δημιουργηθεί ένα νέο πρόγραμμα σε C με τον Builder εκτελούμε τις ακόλουθες εντολές μενού:

- > File
- > New

στο παράθυρο το οποίο ανοίγει επιλέγουμε (εικ. 1)

- > Console Wizard

Εικ. 1

ΔΗΜΙΟΥΡΓΙΑ ΠΡΟΓΡΑΜΜΑΤΟΣ

Στο παράθυρο του **Console Wizard** επιλέγουμε σαν **Source Type C** (ενημερώνουμε τον Builder ότι το πρόγραμμά μας θα είναι σε C) (εικ. 2)

Εικ. 2

Το αποτέλεσμα των ενεργειών μας φαίνεται στην εικόνα

Έχει ανοίξει ο code Editor στην γραμμή τίτλου του οποίου υπάρχει το προσωρινό όνομα του προγράμματός μας το οποίο είναι Unit1.c

Το πρώτο πρόγραμμα

- Σβήνουμε οτιδήποτε είναι γραμμένο στον code editor και γράφουμε τον κώδικα του πρώτου μας προγράμματος

```

#include <stdio.h>
main()
{
    printf("Τεϊ serron \n");
    printf("Τmima pilroforikis ");
}
  
```

Μεταγλώττιση(1)

- Αρχικά αποθηκεύουμε στο φάκελό μας τον πηγαίο κώδικα προσέχοντας να τον αποθηκεύσουμε σαν αρχείο τύπου C File (.c)
- Στη συνέχεια από το μενού Project επιλέγουμε Compile Unit ή πατάμε Alt+F9



19

Μεταγλώττιση(2)

- Αν δεν υπάρχουν συντακτικά λάθη τότε στο φάκελό μας έχει δημιουργηθεί το αντικείμενο πρόγραμμα (.obj)
- Αν υπάρχουν λάθη τότε ενημερωνόμαστε με το κατάλληλο μήνυμα για το είδος του λάθους και την γραμμή στην οποία υπάρχει



Σύνδεση

- Για να γίνει η σύνδεση και η δημιουργία του εκτελέσιμου αρχείου από το μενού Project επιλέγουμε Build
- Το γίνεται και με την εντολή Make, με τη διαφορά ότι η Make μεταγλωττίζει μόνο τα αρχεία που έχουν αλλάξει από την προηγούμενη σύνδεση



21

Εκτέλεση

- Για να εκτελέσουμε το πρόγραμμα από το μενού Run επιλέγουμε Run ή πατάμε F9 ή χρησιμοποιούμε το αντίστοιχο κουμπί της γραμμής εργαλείων



22

Τα βασικά στοιχεία ενός προγράμματος

```
#include <stdio.h>
/* πρόγραμμα 1 */
main()
{
    printf("Tei serron \n");
    printf("Tmima pliroforikis ");
}
```

← Εντολές προεπεξεργαστή
 ← Σχόλια
 ← Συνάρτηση main
 ← Αρχή τμήματος κώδικα
 ← Συνάρτηση Εξόδου
 ← Το ερωτηματικό συμβολίζει το τέλος μιας εντολής

23

Τα βασικά στοιχεία ενός προγράμματος (2)

- **#include**: Είναι εντολή προεπεξεργαστή. Με την εντολή αυτή περιλαμβάνονται στο πρόγραμμά μας αρχεία κεφαλίδας (πχ. stdio.h)
- **Σχόλια**: Για να βάλουμε σχόλια στη C μπορούμε είτε να τα συμπεριλάβουμε ανάμεσα στους χαρακτήρες /* και */ είτε βάζοντας στην αρχή μιας γραμμής τα //
- **main()**: Είναι συνάρτηση. Όλα τα προγράμματα στη C αποτελούνται από συναρτήσεις. Η main είναι η κύρια συνάρτηση κάθε προγράμματος και η εντολή που βρίσκεται μετά το άγκιστρό της η πρώτη η οποία εκτελείται σε κάθε πρόγραμμα. Δεν μπορεί να υπάρξει πρόγραμμα χωρίς main.

24

Τα βασικά στοιχεία ενός προγράμματος (3)

- Με **άγκιστρα** στη C συμβολίζουμε την αρχή και το τέλος ενός τμήματος κώδικα.
- Το **ερωτηματικό** ; συμβολίζει το τέλος μιας εντολής
- **printf**: Είναι συνάρτηση εξόδου η οποία τυπώνει το κείμενο που υπάρχει μέσα στα εισαγωγικά
- **\n** : Ακολουθία διαφυγής, σημαίνει μετακινήσου στην επόμενη γραμμή
- **ΠΡΟΣΟΧΗ**
Η C σαν γλώσσα είναι **case sensitive** δηλαδή κάνει διάκριση μεταξύ πεζών και κεφαλαίων (π.χ θα είναι λάθος να γράψουμε Main αντί για main). Όλες οι εντολές στη C γράφονται με πεζά

25

Λεξιλόγιο της γλώσσας C

- **Δεσμευμένες λέξεις (reserved words)**
- **Λέξεις κλειδιά (keywords)**
- **Τελεστές (operators)**
- **Αναγνωριστές (identifiers)**

26

Δεσμευμένες λέξεις

Πρέπει να αποφύγεται η χρήση τους ως ονόματα

- **Ονόματα συναρτήσεων** πρώτης βιβλιοθήκης (runtime function names), όπως **printf**, **abs** κ.λ.π.
- **Macro names**. Είναι ονόματα που περιέχονται σε αρχεία κεφαλίδας για ορισμό μακροεντολών, π.χ. **EOF**, **INT_MAX**.
- **Type names**. Είναι ονόματα τύπων σε ορισμένα αρχεία κεφαλίδας, π.χ. **time_t**, **va_list**.
- **Ονόματα εντολών** προεπεξεργαστή (preprocessor). Είναι ονόματα που χρησιμοποιεί προεπεξεργαστής της C και έχουν προκαθορισμένη σημασία, π.χ. **include**, **define**.
- **Ονόματα** που αρχίζουν με το χαρακτήρα υπογράμμισης **_** και έχουν δεύτερο χαρακτήρα τον ίδιο ή κεφαλαίο γράμμα, π.χ. **_DATE_**, **_FILE_**.

27

Λέξεις κλειδιά στην ANSI C:

auto	else	register	union
break	enum	return	unsigned
case	extern	short	void
char	float	signed	volatile
const	for	sizeof	while
continue	goto	static	default
if	struct	do	int
switch	double	long	typedef

28

Αναγνωριστές

Λεκτικές μονάδες που κατασκευάζει ο προγραμματιστής. Αυτές οι λεκτικές μονάδες χρησιμοποιούνται συνήθως ως **ονόματα** που ο προγραμματιστής δίνει σε δικές του κατασκευές, όπως **μεταβλητές**, **σταθερές**, **συναρτήσεις** και δικούς του τύπους δεδομένων. Ένα όνομα προσδιορίζει μοναδικά (uniquely identifies), από το σύνολο των κατασκευών του προγράμματος, την κατασκευή στην οποία αποδόθηκε, εξ ου και το όνομα **αναγνωριστής** (identifier).

29