

ΔΟΜΕΣ

1. Οι δομές στη C

Μια δομή είναι μια συλλογή μιας ή περισσότερων μεταβλητών ομαδοποιημένων με ένα όνομα για εύκολο χειρισμό. Οι μεταβλητές σε μια δομή, σε αντίθεση με τον πίνακα, μπορούν να είναι διαφορετικών τύπων. Μια δομή μπορεί να περιέχει οποιουδήποτε τύπους δεδομένων της 'C' (π.χ. float, int, double, char κτλ), συμπεριλαμβανομένων και άλλων δομών και πινάκων.

1.1 Ορισμός και δήλωση δομών

Η λέξη κλειδί 'struct' χρησιμοποιείται για να δηλώσει δομές και προσδιορίζει την αρχή του ορισμού μιας δομής. Ακολουθείται από μια ετικέτα που είναι το όνομα που δίνεται στην δομή. Ακολουθούν τα μέλη της δομής μέσα σε άγκιστρα.

```
struct onoma_domis
{
    typos_metablitis onoma1;
    typos_metablitis onoma2;
    typos_metablitis onoma3;
};
```

Για να χρησιμοποιηθεί το παρακάτω 'struct' θα πρέπει να δηλώσουμε μεταβλητές οι οποίες είναι του τύπου της δομής που έχει δημιουργήσει ο προγραμματιστής struct onoma_domis onoma_metablitis;

ΠΑΡΑΔΕΙΓΜΑ

```
#include <stdio.h>

main ()
{
    struct coord{ /* Dimioyrgia domis coord */ int x;
    int y;
};

struct coord value1, value2;

value1.x=35; /*Ekxwrisi timis sto stoixeio x tis metablitis value1*/
/* pou einai typos domis coord */ value1.y=54; /*Ekxwrisi timis sto stoixeio y tis
metablitis value1*/
/* pou einai typos domis coord */
}
```

Ένα βασικό πλεονεκτήματα των δομών είναι ότι μπορούν να αντιγραφούν δεδομένα μεταξύ των δομών του ίδιου τύπου με μια απλή πρόταση.

ΠΑΡΑΔΕΙΓΜΑ

```
#include <stdio.h>

main ()
{
    struct    time{ /* Dimioyrgia domis time */ int    year;
    int hour; int minutes; int seconds;
    };

    struct time birth_ofday={1980,11,55,30}; /* Apodosi timwn se kathe */
    /* stoixeio tis domis */
    stuct time new_birth;

    new_birth=birth_ofday;

    printf("year of birth is %d\n",new_birth.year); printf("time of birth is %d:%d:d\n",
    new_birth.hour, new_birth.minutes,new_birth.seconds,);
}
```

Στην οθόνη του υπολογιστή εμφανίζεται το παρακάτω αποτέλεσμα

```
Year of birth is 1980
Time of birth is 11:55:30
```

1.2 Πρόσβαση Μελών στις Δομές

Τα μεμονωμένα μέλη των δομών μπορούν να χρησιμοποιηθούν όπως και οι άλλες μεταβλητές του ίδιου τύπου. Τα μέλη των δομών προσπελούνται με χρήση του τελεστή μέλους δομής (.) που ονομάζεται και τελεστής τελείας. Ο τελεστής αυτός τοποθετείται ανάμεσα στη μεταβλητή της δομής και στο όνομα του μέλους.

ΠΑΡΑΔΕΙΓΜΑ

```
#include <stdio.h>

main ()
{
    struct    coord{ /* Dimioyrgia domis coord */
    int    x;
    int    y;
    };

    struct coord value1, value2;

    value1.x=35; /*Ekxwrisi timis sto stoixeio x tis metablitis value1*/
    /* poy einai tyroy domis coord */
    value1.y=54; /*Ekxwrisi timis sto stoixeio y tis metablitis value1*/
    /* poy einai tyroy domis coord
```

Ένα βασικό πλεονεκτήματα των δομών είναι ότι μπορούν να αντιγραφούν δεδομένα μεταξύ των δομών του ίδιου τύπου με μια απλή πρόταση

ΠΑΡΑΔΕΙΓΜΑ

```
#include <stdio.h>

main ()
{
    struct coord{ /* Dimiourgia domis coord */
        int x;
        int y;
    };

    struct coord value1, value2;

    value1.x=35; /*Ekwxwrisi timis sto stoixeio x tis metablitis value1*/
    /* pou einai tyroy domis coord */
    value1.y=54; /*Ekwxwrisi timis sto stoixeio y tis metablitis value1*/
    /* pou einai tyroy domis coord */

    value2.x =value1.x;
    value2.y= value1.y;
    /* Oi parapanw ekwxriseis einai isodynamoi me to parakatw */ value1=value1;
}
```

1.3 Πολυπλοκότερες Δομές

Μια δομή της 'C' μπορεί να περιέχει οποιουδήποτε τύπους δεδομένων της 'C'. Για παράδειγμα μία δομή μπορεί να περιέχει άλλες δομές. Το προηγούμενο παράδειγμα μπορεί να επεκταθεί για να δείξει αυτό το σημείο.

ΠΑΡΑΔΕΙΓΜΑ

```
#include <stdio.h>

main ()
{
    struct coord{ /* Dimiourgia domis coord */
        int x;
        int y;
    };

    struct rectangle{
        struct coord topleft;
        struct coord bottomwrt;
    };

    struct rectangle mybox;

    mybox.topleft.x=35; mybox.topleft.y=45;

    mybox.bottomwrt.x = 78;
    mybox.bottomwrt.y = 11;
}
```

Μπορούν να ορισθούν δομές που περιλαμβάνουν πίνακες οποιουδήποτε τύπου (π.χ. char, int, double κτλ). Η προσπέλαση των μεμονωμένων στοιχείων της διάταξης που είναι μέλη των δομών πραγματοποιείται, χρησιμοποιώντας ένα συνδυασμό του τελεστή μέλους και των δεικτών του πίνακα.

ΠΑΡΑΔΕΙΓΜΑ

```
#include <stdio.h>

main ()
{
    struct entry{ /* Dimioyrgia domis entry */
    char fname[20];
    char iname[20];
    char phone[8];
    };

    struct entry list[999];

    for(i=0;i<999;i++)
    {
        printf("Enter the first name\n");
        scanf("%s",list[i].fname);
        printf("Enter the surname\n");
        scanf("%s",list[i].iname);
        printf("Enter the phone number 123-4567\n");
        scanf("%s",list[i].phone);
    }
}
```

Το παραπάνω πρόγραμμα δηλώνει μια διάταξη με το όνομα 'list' η οποία περιέχει 1000 στοιχεία. Κάθε στοιχείο είναι μια δομή τύπου entry και αναγνωρίζεται από τον δείκτη, όπως και τα στοιχεία κάθε άλλης δομής. Καθεμία από αυτές τις δομές έχει τρία στοιχεία καθένα των οποίων είναι μία διάταξη τύπου char. Ολόκληρη αυτή η σύνθεση δημιουργία απεικονίζεται στο παρακάτω σχήμα.

1.4 Δομές και Δείκτες

Υπάρχει η δυνατότητα της χρήσης δεικτών με τις δομές. Μπορείτε να χρησιμοποιήσετε δείκτες ως μέλη δομών και μπορείτε επίσης να δηλώσετε δείκτες ως δομές. Τα μέλη δείκτες μιας δομής δηλώνονται με τον ίδιο τρόπο που δηλώνονται οι δείκτες χρησιμοποιώντας τον τελεστή της έμμεσης διευθυνσιοδότησης (*).

ΠΑΡΑΔΕΙΓΜΑ

```
#include <stdio.h>

main ()
{
    struct data{
    int *value;
    int *rate;
    };
}
```

```
struct data first;
int cost, interest;

first.value=&cost;
first.rate=&interest;
}
```

1.5 Δομές και Συναρτήσεις

Οι δομές μπορούν να διοχετεύονται σε συναρτήσεις διοχετεύοντας τα μεμονωμένα μέλη της δομής, διοχετεύοντας μια ολόκληρη δομή ή διοχετεύοντας ένα δείκτη προς μια δομή. Όταν διοχετεύονται δομές ή μεμονωμένα μέλη δομής σε μια συνάρτηση διοχετεύονται μέσω τιμής. Ως εκ τούτου τα μέλη μιας καλούσας δομής δεν μπορούν να τροποποιηθούν από την κληθείσα συνάρτηση. Για να διοχετεύσετε μια δομή μέσω αναφοράς θα πρέπει να διοχετευθεί τη διεύθυνση της μεταβλητής δομής. Οι πίνακες δομών-διοχετεύονται αυτομάτως μέσω αναφοράς.

