

Φροντιστήριο - GCC Errors
20/10/2015

Στόχος του φροντιστηρίου είναι η εξοικείωση με τα μηνύματα που εμφανίζει ο μεταφραστής gcc όταν υπάρχουν συντακτικά λάθη στο πρόγραμμα μας. Έστω για παράδειγμα το παρακάτω πρόγραμμα ([κατεβάστε το από εδώ](#)):

```
1 #include <stdio.h>
2
3 int main(void) {
4     float a, b, c;
5
6     a = 0.5;
7     b = 4.8;
8
9     c = mul(a, b);
10
11    if(c < 10) {
12        printf("Smaller than 10\n");
13    }
14    else {
15        printf("Greater than 10\n");
16    }
17    return 0
18 }
19
20 float mul(float num1, float num2) {
21     return num1 * num2;
22 }
```

Σχήμα 1: Εκτύπωση αρχικού κώδικα

Όταν μεταφραστεί το πρόγραμμα του Σχ. 1 θα βγει το ακόλουθο μήνυμα:

```
fix.c: In function 'main':
fix.c:18: error: expected ';' before '}' token
fix.c:20: error: conflicting types for 'mul'
fix.c:9 : note: previous implicit declaration of 'mul' was here
fix.c:22: error: expected declaration or statement at end of input
```

Τα μηνύματα του gcc έχουν την εξής μορφή:

<όνομα αρχείου>: <όνομα συνάρτησης>
<όνομα αρχείου>: <αριθμός γραμμής>: <τύπος μηνύματος>: <μήνυμα>

Στο συγκεκριμένο παράδειγμα έχουμε 3 μηνύματα με τύπο *error* και 1 μήνυμα με τύπο *note*. Το καλύτερο είναι να αρχίσουμε να διορθώνουμε τα λάθη με την σειρά την οποία μας τα εμφανίζει ο μεταφραστής. Το πρώτο μήνυμα *error* αφορά την γραμμή 18 του αρχείου μας και συγκεκριμένα μας ενημερώνει ότι στην γραμμή αυτή λείπει ο χαρακτήρας ;. Αν κοιτάξουμε το πρόγραμμα θα δούμε πράγματι ότι στην γραμμή 18 θα έπρεπε να υπάρχει το σύμβολο ; πριν το } καθώς έχουμε ξεχάσει να το γράψουμε στην αμέσως προηγούμενη εντολή (γραμμή 17). Αφού διορθώσουμε το λάθος αυτό μεταφράζουμε ξανά το πρόγραμμα και τώρα έχουμε το ακόλουθο μήνυμα:

```
fix.c: In function 'main':
fix.c:20: error: conflicting types for 'mul'
fix.c:9 : note: previous implicit declaration of 'mul' was here
fix.c:22: error: expected declaration or statement at end of input
```

Παρατηρούμε ότι το επόμενο λάθος είναι στην γραμμή 20, εκεί που καλείται η συνάρτηση *mul*. Η αιτία του λάθους είναι ότι δεν έχει δηλωθεί η συνάρτηση *mul* πριν την κλήση της στην *main*, οπότε ο μεταφραστής δεν γνωρίζει την συγκεκριμένη συνάρτηση. Όταν συμβαίνει αυτό (στην γραμμή 9) ο μεταφραστής υποθέτει ότι όλα της τα ορίσματα της συνάρτησης καθώς και ο τύπος της επιστρεφόμενης μεταβλητής είναι είναι ακέραιοι (*int*). Στην συνέχεια στην γραμμή 20 βλέπει στον ορισμό της συνάρτησης ότι τα ορίσματα και η επιστρεφόμενη μεταβλητή είναι τύπου *float* εξ ου το *error* και το *note*. Η λύση στο πρόβλημα αυτό είναι να ορίσουμε το πρωτότυπο της συνάρτησης *mul* πριν την κλήση της συνάρτησης, π.χ. μετά τις δηλώσεις των *include* ή μέσα στην *main*. Επίσης μπορούμε να μεταφέρουμε

την δήλωση της συνάρτησης *mul* πάνω από την *main*. Στο παράδειγμα αυτό επιλέγουμε την δεύτερη λύση, έτσι μετά την αλλαγή αυτή ο κώδικας θα είναι:

```
1 #include <stdio.h>
2
3 float mul(float num1, float num2) {
4     return num1 * num2;
5 }
6
7 int main(void) {
8     float a, b, c;
9
10    a = 0.5;
11    b = 4.8;
12
13    c = mul(a, b);
14
15    if(c < 10) {
16        printf("Smaller than 10\n");
17    }
18    else {
19        printf("Greater than 10\n");
20    }
21    return 0;
22 }
```

Σχήμα 2: Εκτύπωση τροποποιημένου κώδικα

Μετά την νέα μετάφραση θα εμφανιστεί το μήνυμα:

```
fix.c: In function 'main':
fix.c:22: error: expected declaration or statement at end of input
```

Το τελευταίο λάθος μας ενημερώνει ότι ο μεταφραστής περιμένει κάποια δήλωση μεταβλητής ή κάποια εντολή κώδικα στην γραμμή 22 αντί για το τέλος του προγράμματος. Αυτό συμβαίνει διότι κάποια αγκύλη δεν έχει κλείσει (`}`), οπότε για τον μεταφραστή το ο κώδικας του προγράμματος δεν τελειώνει με σωστό τρόπο. Πράγματι έχει παραληφθεί η αγκύλη `}` που δηλώνει το τέλος του μπλόκ κώδικα ο οποίος αρχίζει στην γραμμή 18. Μετά την διόρθωση και αυτού του σφάλματος το πρόγραμμα μπορεί μεταγλωττιστεί χωρίς λάθη.