

**Τεχνολογικό Πανεπιστήμιο Κύπρου  
Σχολή Μηχανικής και Τεχνολογίας   
Τμήμα Ηλεκτρολόγων Μηχανικών, Μηχανικών Υπολογιστών και Πληροφορικής**

CEI 222: Αρχές Γλωσσών Προγραμματισμού και Μεταγλωττιστές

Εργαστήριο 2ο   
Φθινοπωρινό Εξάμηνο 2021  
  
  
  
**Όνομα Διδάσκων: Δρ. Ε. Κακουλλή  
Ονόματα Φοιτητών: Σωτήρης Βασιλειάδης,** **19613**

**Μιχαήλ-Άγγελος Δήμου, 19753**

**Κωνσταντίνος Κωνσταντίνου,** **20284**

**Γιώργος Τσοβίλης, 19971**

**Ημερομηνία Παράδοσης: 01/11/21**

Κατανομή ευθυνών:

* Ο Μιχαήλ-Άγγελος Δήμου σύγγραψε τα input samples για το πρόγραμμα, μάζεψε τα screenshots που χρειάστηκαν και βοήθησε με την συγγραφή της αναφοράς.
* Ο Σωτήρης Βασιλειάδης, ο Κωνσταντίνος Κωνσταντίνου, ο Μιχαήλ-Άγγελος Δήμου και ο Γιώργος Τσοβίλης διάβασαν εις βάθος το πρόβλημα, και αφού κατανόησαν σαν ομάδα τί ζητούσε η άσκηση, σκέφτηκαν τη δομή του κώδικα.
* Ο Κωνσταντίνος Κωνσταντίνου και ο Γιώργος Τσοβίλης ετοίμασαν την αναφορά, και διόρθωσαν τα παραπάνω συντακτικά/ορθογραφικά λάθη που εμφανίζονταν κατά τη διάρκεια συγγραφής του κώδικα.
* Ομαδικώς μελετήσαμε περεταίρω το πώς λειτουργεί η γλώσσα C- και το εργαλείο bison και δουλέψαμε μαζί για να γράψουμε τον τελικό κώδικα, κυρίως ο Σωτήρης Βασιλειάδης.

Προβλήματα που αντιμετωπίσαμε:

* Τυπώνοντας το συντακτικό δέντρο στην σωστή σειρά.
* Διαβάζοντας το όνομα μιας μεταβλητής.

Πως αντιμετωπίσαμε τα προβλήματα αυτά:

* Χρησιμοποιώντας ένα απλό string vector, προσθέταμε τα σημεία που περνούσε ο συντακτικός αναλυτής, και όταν φτάσει στο τέλος του προγράμματος τα τυπώνουμε ανάποδα για να έχουμε την σωστή σειρά.
* Λόγο της μορφής Union να μην δέχεται string, έπρεπε να χρησιμοποιήσουμε τύπο Char\* για να πάρουμε ολόκληρο το όνομα μιας μεταβλητής/συνάρτησης. Κατά την μεταγλώττιση υπάρχει warning (όχι error) λόγο της μετατροπής από Char\* σε String, αλλά δεν μας ενοχλεί και δουλεύει σωστά. Επίσης στην πραγματικότητα δεν θα τυπώναμε έτσι το δέντρο στα επόμενα step άρα ακόμη ένας λόγος να μην μας ενοχλεί. Στα commands που χρησιμοποιούμε απλά προσθέτουμε το -w.

Σύντομη περιγραφή κώδικα :

* Χρησιμοποιώντας τον λεκτικό αναλυτή flex για αναγνώριση λεκτικών εκφράσεων, και το εργαλείο bison για τη δημιουργία κανόνων, το πρόγραμμα δέχεται σαν input ένα αρχείο το οποίο ο λεκτικός αναλυτής αναγνωρίζει τις λεκτικές εκφράσεις, τις οποίες στέλνει στο bison, και μετά ανάλογα με τον κανόνα που είναι σύμφωνη αυτή η έκφραση, εμφανίζει το ανάλογο μήνυμα. Επίσης μπορούμε να γράψουμε κώδικα απευθείας από το terminal και θα μας βγάλει το συντακτικό δέντρο.

**Input Samples:**

**Input 1:**

/\*A program\*/

void main(void)

{

int x; int y;

x = input();

y = x + 5;

}

**Input 2:**

/\*A program\*/

void main(void)

{

int x[10]; int y;

y=2;

if(y==2){

x = input();

y = x + 5;

}

}

**Input 3:**

/\*A program\*/

void main(void)

{

int x[10]; int y;

y=2;

while(y!=8){

x[y] = y;

y = y + 1;

}

}

**Output 1 :**

Text

Description automatically generated

**Output 2 :**

**Text

Description automatically generated**

**Text

Description automatically generated**

**Output 3 :**

**Text

Description automatically generated**

**Text

Description automatically generated**