

**Τμήμα Πληροφορικής Α.Π.Θ.**  
Προαιρετική Εργασία στο μάθημα  
Γλώσσες Προγραμματισμού και Μεταγλωττιστές  
Ακ. Έτος: 2023-2024

**Bonus βαθμού ή απαλλαγή από εξέταση**

Η προαιρετική εργασία εκπονείται **αποκλειστικά σε ατομική βάση** και το όφελος που θα έχουν οι φοιτητές ανάλογα με την περίπτωση είναι:

- **απαλλαγή από την εξέταση του Ιουνίου/Σεπτεμβρίου**  
Οι φοιτητές που θα εκπονήσουν απαλλακτική εργασία θα προσέλθουν στην εξέταση μόνο για να δηλώσουν στο γραπτό ότι παρέδωσαν ή προτίθενται να παραδώσουν και να βαθμολογηθούν με βάση την εργασία. Η δυνατότητα απαλλακτικής εργασίας ισχύει για την εξεταστική του Ιουνίου αλλά και για την εξεταστική Σεπτεμβρίου. Όσοι παραδώσουν απαλλακτική εργασία και το πρόγραμμα τους εκτελείται σωστά, αυτοί θα βαθμολογούνται με ΑΡΙΣΤΑ (10).
- **δυνατότητα να παραδοθεί μέρος μόνο της εργασίας**  
Οι φοιτητές που θα επιδείξουν ολοκληρωμένη λεξική, συντακτική και σημασιολογική ανάλυση με πίνακα συμβόλων, των οποίων ο μεταγλωττιστής θα εκτυπώνει το συντακτικό δέντρο του πηγαίου προγράμματος θα δικαιούνται 30% bonus στον τελικό βαθμό της εξέτασης του Ιουνίου ή του Σεπτεμβρίου. Μπορεί κάποιος να γράψει στην εξεταστική του Ιουνίου και αν θέλει να κρατηθεί η επίδοση του για την εξεταστική Σεπτεμβρίου, έτσι ώστε αν παραδοθεί τότε η εργασία να συμπληρωθεί ο βαθμός με το bonus 30% στο βαθμό της τελευταίας εξέτασης που έχει δώσει.

**Αξιολόγηση εργασίας**

Για να βαθμολογηθεί η εργασία, θα πρέπει **να παρουσιαστεί στο γραφείο του διδάσκοντα**. Η συνάντηση θα κανονιστεί σε συνεννόηση με τους διδάσκοντες μετά από την εξέταση του μαθήματος για την εξεταστική του Ιουνίου και ομοίως για την εξεταστική του Σεπτεμβρίου. Κατά την παρουσίασή σας θα περιγράψετε τη σχεδίαση του μεταγλωττιστή, θα επιδείξετε σε φορητό υπολογιστή τη λειτουργία του και θα απαντήσετε σε σχετικές διευκρινιστικές ερωτήσεις. Το παραδοτέο για να είναι αποδεκτό πρέπει να περιλαμβάνει:

- σε ηλεκτρονική μορφή
  - i. τον κώδικα της γλώσσας σε πηγαία (source) και εκτελέσιμη μορφή (executable)
  - ii. αρχεία κειμένου ASCII με παραδείγματα προγραμμάτων έτοιμα για μεταγλώττιση και εκτέλεση
- αναφορά με
  - i. συνοπτική περιγραφή της γλώσσας και της υλοποίησης της
  - ii. παραδείγματα και αποτελέσματα από την μεταγλώττιση τους και την εκτέλεση του κώδικα μηχανής

**Αν ο διδάσκων κρίνει ότι το πρόγραμμα, που παραδίνετε δεν είναι αποτέλεσμα προσωπικής εργασίας, τότε δε θα λαμβάνεται υπόψη στον τελικό βαθμό.**

**Υλοποίηση εργασίας**

Ο μεταγλωττιστής μπορεί να υλοποιηθεί στη γλώσσα προγραμματισμού της προτίμησής σας αρκεί να έχετε στη διάθεσή σας τα απαραίτητα εργαλεία (π.χ. flex και yacc ή κλώνους για γλώσσες προγραμματισμού εκτός της C). Τα περισσότερα εργαλεία αυτού του τύπου λειτουργούν όπως περιγράφεται στις σημειώσεις/διαφάνειες/σύγγραμμα μαθήματος και όπως φαίνεται από την υλοποίηση του πρότυπου εκπαιδευτικού μεταγλωττιστή, που ο κώδικάς του (σε γλώσσα C) διατίθεται στα εργαστηριακά που έχουμε κάνει και έχουν αναρτηθεί στην ιστοσελίδα του μαθήματος στο elearning.

Τις τελευταίες εκδόσεις των εργαλείων flex και yacc σε εκτελέσιμη μορφή για windows μπορείτε να τις βρείτε στη διεύθυνση <http://gnuwin32.sourceforge.net/packages.html>. Tutorial για τη χρήση τους μπορείτε να βρείτε στη διεύθυνση <https://developer.ibm.com/tutorials/au-lexyacc/> και στη διεύθυνση [https://arcb.csc.ncsu.edu/~mueller/codeopt/codeopt00/y\\_man.pdf](https://arcb.csc.ncsu.edu/~mueller/codeopt/codeopt00/y_man.pdf) καθώς και στο σύγγραμμα του μαθήματος. Ο μεταγλωττιστής που θα κατασκευάσετε θα κάνει μετάφραση στη γλώσσα της μηχανής MIX, που περιγράφεται παρακάτω.

### Σύνταξη Γλώσσας Προγραμματισμού

Η σύνταξη της «ΑΠΛΗΣ ΓΛΩΣΣΑΣ» ορίζεται από τη γραμματική χωρίς συμφραζόμενα:

```
1   program      ::= stmt_seq
2   stmt_seq     ::= stmt_seq ; stmt
3               | stmt
4   stmt         ::= assign_stmt
5               | if_stmt
6               | repeat_stmt
7               | read_stmt
8               | write_stmt
9   assign_stmt  ::= id := exp
10  read_stmt    ::= read exp
11  write_stmt   ::= write id
12  if_stmt      ::= if exp then stmt_seq end
13               | if exp then stmt_seq else stmt_seq end
14  repeat_stmt  ::= repeat stmt_seq until exp
15  factor       ::= ( exp )
16               | number
17               | id
18  term         ::= factor
19               | term * factor
20               | term / factor
21  simple_exp   ::= term
22               | simple_exp + term
23               | simple_exp - term
24  rel_exp      ::= simple_exp
25               | rel_exp < simple_exp
26               | rel_exp = simple_exp
27  exp          ::= rel_exp
```

Για κάθε έναν από τους κανόνες της γραμματικής περιγράφουμε τη σημασία του. Ένα πρόγραμμα (program) (βλ. κανόνα 1) αποτελείται από μία ακολουθία εντολών (stmt\_seq). Μία ακολουθία εντολών είτε περιλαμβάνει μία μόνο εντολή (βλ. κανόνα 3) είτε περιλαμβάνει περισσότερες από μία εντολές που διαχωρίζονται με ';' (βλ. κανόνα 2). Διακρίνουμε πέντε τύπους εντολών (βλ. κανόνες 4, 5, 6, 7 και 8) την εκχώρηση τιμής (assign\_stmt), την εντολή επιλογής (if\_stmt), την εντολή επανάληψης (repeat\_stmt) και τις εντολές ανάγνωσης (read\_stmt) και εκτύπωσης (write\_stmt). Μία εκχώρηση τιμής (βλ. κανόνα 9) αποτελείται από ένα όνομα μεταβλητής (τερματικό σύμβολο), που ακολουθείται από ένα σύμβολο εκχώρησης τιμής (τερματικό) και μία

έκφραση. Η έκφραση (βλ. κανόνες 27 και 24-26) μπορεί να είναι μία απλή αριθμητική έκφραση ή μία λογική έκφραση με χρήση των τελεστών  $<$  ή  $=$  οι οποίοι έχουν τη γνωστή σημασία. Το αποτέλεσμα μιας έκφρασης είναι η αριθμητική της τιμή ή αν αυτή είναι λογική έκφραση η τιμή 1 αν αποτιμάται σε true ή η τιμή 0 αν το αποτέλεσμα είναι false. Η εκχώρηση τιμής έχει επίσης τη συνήθη σημασία: εντοπίζεται η θέση της μεταβλητής με όνομα τη λεξική μονάδα που αντιστοιχεί στο id, στη συνέχεια υπολογίζεται η υποέκφραση που βρίσκεται στα δεξιά του τελεστή := και η τιμή αυτής της υποέκφρασης αποθηκεύεται στη θέση της μεταβλητής με το συγκεκριμένο όνομα. Η τιμή που επιστρέφεται για την εντολή εκχώρησης είναι τελικά η τιμή της υποέκφρασης. Η εντολή if (βλ. κανόνες 12 - 13) έχει επίσης τη συνηθισμένη σημασία: υπολογίζεται η έκφραση της συνθήκης και μία μη μηδενική τιμή προκαλεί τελικά την εκτέλεση της ακολουθίας εντολών (stmt\_seq), που αναφέρεται πρώτη. Στην περίπτωση που η έκφραση της συνθήκης αποτιμάται σε μηδενική τιμή, τότε εκτελείται η δεύτερη ακολουθία εντολών αν βέβαια υπάρχει. Τέλος, η εντολή επανάληψης (repeat\_stmt) έχει επίσης τη συνηθισμένη σημασία.

### Παράδειγμα προγράμματος της Γλώσσας

Το πρόγραμμα που ακολουθεί σε «ΑΠΛΗ ΓΛΩΣΣΑ» υπολογίζει το παραγοντικό ενός αριθμού που διαβάζεται από το πληκτρολόγιο.

```
read x;
if 0 < x then
  fact := 1;
  repeat
    fact := fact * x;
    x := x - 1
  until x = 0;
  write fact
end
```

### Παραγωγή κώδικα στη γλώσσα - στόχο

Στην περίφημη σειρά βιβλίων “The Art of Computer Programming”, που δημοσίευσε ο D.Knuth, χρησιμοποιείται ένας ιδεατός υπολογιστής που αποκαλείται MIX. Ο MIX διαθέτει γλώσσα μηχανής και συμβολικής αναπαράστασης (assembly) των εντολών της.

Η αρχιτεκτονική του MIX είναι μία απλοποιημένη εκδοχή της σχεδίασης των πραγματικών CISC CPUs και η συμβολική γλώσσα του, που λέγεται MIXAL, παρέχει ένα σύνολο βασικών εντολών, που είναι πολύ οικείες σε όποιον έχει εμπειρία προγραμματισμού σε γλώσσα assembly. Ο ορισμός του MIX και της MIXAL είναι αρκετά ισχυρός και πλήρης, ώστε να παρέχει μία ιδεατή πλατφόρμα ανάπτυξης σύνθετων προγραμμάτων, αρκετά κοντά στους πραγματικούς υπολογιστές, χωρίς όμως την πολυπλοκότητα της αρχιτεκτονικής αυτών. Για αυτό το λόγο εξάλλου ο MIX και η MIXAL χρησιμοποιούνται ευρέως ως εκπαιδευτικό περιβάλλον εκμάθησης προγραμματιστικών τεχνικών.

Ένα αρκετά ώριμο project με εργαλεία ανάπτυξης για τη γλώσσα MIXAL είναι το [GNU MIX Development Kit](http://www.gnu.org/software/mdk/manual/mdk.pdf). Αναλυτική περιγραφή της αρχιτεκτονικής της μηχανής MIX και της γλώσσας MIXAL θα βρείτε στο online βιβλίο <http://www.jklp.org/profession/books/mix/index.html> και στον οδηγό (manual), που διατίθεται στη διεύθυνση <http://www.gnu.org/software/mdk/manual/mdk.pdf>. Στην ιστοσελίδα <https://www.mix-emulator.org/#/about> διατίθεται ένας online emulator εκτέλεσης προγραμμάτων MIXAL.

Ο μεταγλωττιστής σας θα πρέπει να δημιουργεί κώδικα MIXAL για προγράμματα της ΑΠΛΗΣ ΓΛΩΣΣΑΣ, που θα μπορεί να εκτελείται σε κάποιο virtual machine ή emulator της MIXAL (διατίθεται virtual machine και το GNU MIX Development Kit).

Πρέπει να τονιστεί ότι η γλώσσα MIXAL υλοποιείται στα προαναφερόμενα εργαλεία με μικρές διαφορές (π.χ. στον ορισμό των σταθερών string) και γι' αυτό πρέπει ο κώδικας που θα δημιουργήσει ο μεταγλωττιστής σας να συμμορφώνεται με τις ιδιαιτερότητες της μηχανής MIX που χρησιμοποιείτε. Συμβουλευτείτε για το σκοπό αυτό τα παραδείγματα των MIXAL προγραμμάτων που συνοδεύουν την πλατφόρμα MIX που έχετε επιλέξει.