

Πανεπιστήμιο Δυτικής Μακεδονίας
Τμήμα Πληροφορικής
Μεταγλωττιστές
Εργασία 4^η
Παράδοση 10/01/2023

Άσκηση 1 [30%]

Αναπτύξτε έναν συντακτικό αναλυτή για τη γεννήτρια BYACC, που θα δέχεται δυαδικούς αριθμούς, θα τους μετατρέπει σε δεκαδικούς και θα τους εμφανίζει. Θα πρέπει να γράψετε τη γραμματική του συντακτικού αναλυτή και τον λεκτικό αναλυτή.

Στην περίπτωση που υπάρχει κάποιο συντακτικό σφάλμα ο αναλυτής θα εμφανίζει το μήνυμα σφάλματος. Μετά την είσοδο ενός δυαδικού αριθμού ο συντακτικός αναλυτής θα εμφανίζει την έκφραση "result = " και το αποτέλεσμα της δεκαδικής αναπαράστασης.

Η είσοδος στο συντακτικό αναλυτή που θα δημιουργήσετε θα δίνεται από το πληκτρολόγιο ή από αρχείο. Μπορείτε να δημιουργήσετε όσες βοηθητικές συναρτήσεις χρειάζεστε.

Άσκηση 2 [70%]

Αναπτύξτε έναν συντακτικό αναλυτή για τη γεννήτρια BYACC, το οποίο για την είσοδο την οποία θα δέχεται από το λεξικό αναλυτή θα δημιουργήσει έναν απλό υπολογιστή ο οποίος θα αναγνωρίζει τις πράξεις =, +, -, *, /, και ^ (ύψωση σε δύναμη) και θα τους δίνει τη σωστή προτεραιότητα, αναγνωρίζοντας παρενθέσεις. Οι αριθμοί μπορούν να είναι ακέραιοι ή πραγματικοί. Επιπρόσθετα ο συντακτικός αναλυτής θα πρέπει να αναγνωρίζει τις **μεταβλητές** και εφόσον βρίσκονται μέσα σε μία αριθμητική παράσταση να συμπεριλαμβάνει την τιμή που μπορεί να έχουν. Οι μεταβλητές θα μπορούν να αποτελούνται από ένα ή περισσότερα σύμβολα (π.χ. str, xy, j).

Ο λεκτικός αναλυτής θα αναγνωρίζει μόνο τα λεξήματα και θα επιστρέφει τα τερματικά και τις τιμές τους στο συντακτικό αναλυτή μέσω της μεταβλητής `yyval`. Ο λεκτικός αναλυτής θα πρέπει να επιστρέφει είτε αριθμητικές τιμές είτε αλφαριθμητικές (για την περίπτωση των μεταβλητών). Συνεπώς, θα πρέπει να ορίσετε εντός του συντακτικού αναλυτή μία **union** δομή με δύο πεδία, ένα δείκτη σε χαρακτήρα και ένα `double` όπως παρουσιάζεται παρακάτω, για να μπορεί ο συντακτικός αναλυτής να δέχεται διαφορετικού τύπου τιμές.

```
%union {  
    double ydouble;  
    char *ystr;  
}
```

Επιπρόσθετα, ο συντακτικός αναλυτής θα πρέπει να έχει μία δομή τύπου **struct** στην οποία θα αποθηκεύονται οι μεταβλητές και οι τιμές τους προκειμένου να μπορούν μετά να χρησιμοποιηθούν στον υπολογισμό της αριθμητικής παράστασης. Μπορείτε να δημιουργήσετε έναν πίνακα (π.χ. 100 θέσεων) τύπου δομής που έχετε ορίσει. Συνεπώς, θα μπορείτε σε κάθε θέση του πίνακα να αποθηκεύετε και ένα στιγμιότυπο της δομής που θα

αντιστοιχεί και μία μεταβλητή. Επιπλέον, θα πρέπει να δημιουργήσετε μία συνάρτηση **int lookup(char *str)**, που θα επιστρέφει τη θέση στη δομή που εντοπίστηκε η μεταβλητή εάν υπάρχει διαφορετικά την τιμή -1 και μία συνάρτηση **void init()** η οποία θα αρχικοποιεί τη δομή.

Στην περίπτωση που υπάρχει κάποιο συντακτικό σφάλμα ο αναλυτής θα εμφανίζει το μήνυμα σφάλματος. Μετά την είσοδο μιας αριθμητικής παράστασης ο συντακτικός αναλυτής θα εμφανίζει (όπως φαίνεται παρακάτω) την έκφραση "result =" και το αποτέλεσμα της αριθμητικής παράστασης.

Η είσοδος στο συντακτικό αναλυτή που θα δημιουργήσετε θα δίνεται από το πληκτρολόγιο ή από αρχείο. Μπορείτε να δημιουργήσετε όσες βοηθητικές συναρτήσεις χρειάζεστε.

```
a=5
b=10
a+b
result = 15.000000
5+a*2-(4+6)
result = 5.000000
3*(2+b)/3-2
result = 10.000000
-7=y
syntax error
```

Εικόνα 1 Παράδειγμα εκτέλεσης συντακτικού αναλυτή

Χρήσιμες Πληροφορίες:

- Η εργασία είναι **ομαδική**. Κάθε ομάδα αποτελείται από 2 άτομα.
- Η αντιγραφή ανιχνεύεται και τιμωρείται.
- **Παράδοση μέσω eclass κάνοντας upload στην ενότητα Εργασία 4. Upload θα κάνει μόνο το ένα από τα δύο άτομα της ομάδας**
- Θα παραδώσετε ένα zip αρχείο το οποίο θα περιέχει τις υλοποιήσεις και των δύο ασκήσεων.
- Το όνομα του αρχείου που θα παραδώσετε θα έχει το παρακάτω format:
 - **Επίθετο1_Όνομα1_ΑριθμόςΜητρώου1_Επίθετο2_Όνομα2_ΑριθμόςΜητρώου2**
- Η εργασία θα πρέπει να παραδοθεί μέχρι τις 10/01/2023 και ώρα 23:59.