

Εκφώνηση(ΘΕΜΑ 5): Έστω το αλφάβητο A, B, C, D, E, F, G, H. Σε ένα υποσύνολο φυσικής γλώσσας, τα ονόματα σημείων ορίζονται ως η παράθεση ενός μόνο συμβόλου, τα ονόματα ευθειών ορίζονται ως η παράθεση δύο συμβόλων, τα ονόματα τριγώνων ορίζονται ως η παράθεση τριών συμβόλων, κ.ο.κ, έως και την περίπτωση οκταγώνων. Δεν επιτρέπονται επαναλήψεις συμβόλων. Να γραφεί πρόγραμμα Flex που θα αναλύει προτάσεις της μορφής «τρίγωνο BCD», «τετράγωνο BCDA», κ.ο.κ. και θα αποδέχεται μόνο τους σωστούς ορισμούς. Παραδείγματα λάθος ορισμών είναι «τετράγωνο AB», «τρίγωνο AAD», «γωνία BC».

Στο συγκεκριμένο θέμα απαιτείτε η κατασκευή ενός προγράμματος flex το οποίο να αναγνωρίζει συγκεκριμένες εκφράσεις βάση των κανόνων της εκφώνησης.

ΛΥΣΗ:

Στα αρχεία thema_5.txt και thema5.l το πρώτο είναι σε .txt μορφή για να μπορείτε να δείτε το κώδικα σε μορφή κειμένου και το .l αρχείο είναι το flex.

Επεξήγηση κώδικα:

1^ο Τμήμα flex κώδικα: Αυτό το τμήμα είναι κενό στο κομμάτι του αυτούσιου κώδικα C, καθώς στην υλοποίησή μου δεν χρειάστηκα κάποια βιβλιοθήκη της C κ.λπ. και έχω δηλώσει στο όνομα <<ID>> τα κεφαλαία γράμματα της αγγλικής αλφαβήτου από Α έως Η με την έκφραση <<[A-H]>> και στο όνομα <<SHAPES>> τις περιπτώσεις πιθανών αποδεκτών εκφράσεων κάνοντας χρήση του ονόματος <<ID>> από πριν. π.χ. το <<"line"><{ID}>{ID}>> υποδηλώνει την ύπαρξη της ακολουθίας χαρακτήρων <<line>> ακολουθούμενη χωρίς κενό μεταξύ τους από δύο γράμματα που ανήκουν στο όνομα <<ID>> δηλαδή από Α έως Η της αγγλικής αλφαβήτου.

2^ο Τμήμα flex κώδικα: Σε αυτό το τμήμα έχω δημιουργήσει **3 κανόνες** για την λειτουργικότητα του προγράμματος. **Ο πρώτος** ενεργοποιείται όταν βρεθεί έκφραση που υπόκειται στο όνομα <<SHAPE>>, τότε θα ελέγχει αν υπάρχει επανάληψη συμβόλων <<ID>> δηλαδή τα κεφαλαία γράμματα του αγγλικού αλφαβήτου πρέπει να είναι μοναδικά σε κάθε έκφραση και όχι να επαναλαμβάνονται(π.χ το <<lineAA>> είναι μη αποδεκτό) αυτό επιτυγχάνεται με χρήση δύο <<for loops>> τα οποία τσεκάρουν αν υπάρχουν δύο ίδιοι χαρακτήρες και αν ναι τότε κάνουν την έκφραση reject(συνάρτηση του flex) αλλιώς αναγνωρίζεται ως <<Valid Expression>>. **Ο δεύτερος** κανόνας υπόκειται στην περίπτωση όπου δεν δοθεί σωστή πρώτη έκφραση(π.χ. <<dsquare>> αντί για <<square>>) και τότε θα απορρίψει την έκφραση λέγοντας <<Invalid Expression Because of:....>>. **Ο τρίτος** υπόκειται στην περίπτωση όπου χρήστης δώσει ένα σύμβολο ή κάποιο χαρακτήρα που δεν αναγνωρίζεται(π.χ. <<!>>) όπου και τότε θα απορριφθεί από το πρόγραμμα με μήνυμα <<Unrecognized Character:....>>.

3^ο Τμήμα flex κώδικα: Είναι αυτό που περιέχει τη <<main()>> και το <<yylex()>> περιέχει τις εντολές που μας έχουν διδαχθεί στις παραδόσεις για να δέχεται το <<input>>(την έκφραση από το χρήστη) και να καλεί με το <<yylex()>> το κώδικα που έχουμε γράψει στα πάνω τμήματα.

(Παρατήρηση: Ο δεύτερος κανόνας είναι και αυτός που μου δίνει τη δυνατότητα να παράγω αποδεκτές εκφράσεις από μη-αποδεκτές όπως λέω παρακάτω στην περιγραφή λειτουργίας)

Περιγραφή Λειτουργίας:

Κατά την εκκίνηση του προγράμματος το τερματικό περιμένει την εισαγωγή της έκφρασης από τον χρήστη και υπάρχουν τα εξής ενδεχόμενα:

a)Η έκφραση να γίνει αποδεκτή

```
lineAC
Valid Expression: lineAC
```

b)Η έκφραση να μην γίνει αποδεκτή να δοθεί ο λόγος που δεν είναι και να παραχθεί η αποδεκτή εκδοχή της αφαιρώντας τον ή τους λάθος χαρακτήρες

```
hexABCDEGH
Valid Expression: hexABCDEG
Invalid Expression because of: H
```

```
123angleA
Unrecognized Character: 1
Unrecognized Character: 2
Unrecognized Character: 3
Valid Expression: angleA
```

c) Η έκφραση να μην αναγνωριστεί καθόλου λόγο λάθος σύνταξης

```
!square
Unrecognized Character: !
Unrecognized Character: s
Unrecognized Character: q
Unrecognized Character: u
Unrecognized Character: a
Unrecognized Character: r
Unrecognized Character: e
```

Παραδοχές στην υλοποίηση:

- Ως Unrecognized Characters μετράνε οι αριθμοί, τα σύμβολα του πληκτρολογίου και οποιοσδήποτε χαρακτήρας άλλης γλώσσας εκτός της αγγλικής
- Χρησιμοποίησα αγγλική γλώσσα στην υλοποίηση, δηλαδή όλα τα μνήματα και οι εκφράσεις που υποστηρίζονται είναι μόνο στα αγγλικά

- Για λόγους ευκολίας έδωσα υποκοριστικά σε κάποια σχήματα στο όνομα <<SHAPES>> αναλυτικά είναι: angle = γωνία , line = γραμμή , trio = τρίγωνο , square = τετράγωνο , pent = πεντάγωνο , hex = εξάγωνο , hept = επτάγωνο , oct = οκτάγωνο
- Υπάρχει ευαισθησία όσο αφορά τα κενά για παράδειγμα η έκφραση <<lineAB>> είναι αποδεκτή αλλά η έκφραση <<lineA B>> δεν είναι όπως ούτε η << lineAB>>.

Οδηγίες Χρήσης/Manual:

- Ανοίγουμε το τερματικό και καθοδηγούμαστε στο φάκελο όπου βρίσκεται το αρχείο thema5.l
- Εκτελούμαι τις εξής εντολές με τη ροή που δείχνουν τα βελάκια(flex thema5.l → gcc lex.yy.c -lfl → ./a.out).
- Δίνουμε εκφράσεις στο πρόγραμμα και στο τερματικό μας αναγράφονται τα αντίστοιχα μηνύματα.
- Για να κλείσουμε την εφαρμογή πατάμε την ίδια στιγμή ctrl+D στο πληκτρολόγιο μας.