

Τμήμα: Ηλεκτρολόγων Μηχανικών & Μηχανικών Υπολογιστών
Μάθημα: Εργαστήριο Τεχνητής Νοημοσύνης.
Διδάσκοντες: Καθηγ. Μανόλης Μαρακάκης,
Δρ, Κονδυλακης Χαρίδημος
Ημερομηνία παράδοσης 8 Μαΐου 2022
Εαρινό εξάμηνο 2021-22

Εργασία 3

Άσκηση 1

Να γράψετε κατηγορημα `add-end-diff-lists(Elem, DiffList, EndDiffList, NewDiffList, EndNewDiffList)` το οποίο είναι αληθές εάν `Elem` είναι ένα στοιχείο το οποίο θέλουμε να προσθέσουμε στο τέλος της ανοικτής λίστας `DiffList`, `EndDiffList` είναι ο δείκτης στο τέλος της ανοικτής λίστας. Επιστρέφει τη νέα ανοικτή λίστα `NewDiffList` και τον δείκτη `EndNewDiffList` στο τέλος της νέας ανοικτής λίστας. *Σημείωση:* Το πρόγραμμα σας δεν πρέπει να έχει αναδρομή για να προσθέσει το στοιχείο `Elem` στο τέλος της ανοικτής λίστας `DiffList`. Να εξετάσετε το πρόγραμμα σας για το στόχο.

?- `add-end-diff-lists(1,[a,b,c,d|EndDL],EndDL,NewDL,EndNewDL).`

`EndDL = [1|EndNewDL],`

`NewDL = [a,b,c,d,1|EndNewDL] ? yes`

μονάδες.1.0

Άσκηση 2

Θεωρήσατε την παρακάτω *Γραμματική Οριστικών Προτάσεων (ΓΟΠ)* «Definite Clause Grammar (DCG)».

`S → NP VP`

`NP → Det N`

`VP → V NP`

`VP → V`

`Det → a, the, 'τίποτα'`

`N → dog, cat, boy, girl`

`V → chased, saw, said, believed`

Να γράψετε κατηγορημα `sentence(S, A)` το οποίο είναι αληθές εάν `S` είναι μια λίστα λέξεων και `A` μια άδεια λίστα. Η Prolog επιστρέφει `yes` εάν η λίστα `S` αποτελεί μια σωστή πρόταση σύμφωνα με την παραπάνω σύνταξη που δίνεται με την Γραμματική Οριστικών Προτάσεων (ΓΟΠ) «Definite Clause Grammar (DCG)». Εάν το `S` είναι μη δεσμευμένη μεταβλητή να σχηματίζει όλες τις συντακτικά σωστές προτάσεις σύμφωνα με την παραπάνω σύνταξη και το λεξικό που δίνεται. **Να μην χρησιμοποιήσετε τη γραμματική της Prolog για DCG κανόνες αλλά να γράψετε εσείς τους παραπάνω DCG κανόνες σε Prolog κανόνες (προτάσεις).** Για την υλοποίηση σας να χρησιμοποιήσετε την ιδέα με τις *λίστες διαφοράς (difference lists)*. Τα ονόματα των μεταβλητών που θα χρησιμοποιήσετε να έχουν νόημα, π.χ. `InputPhrase` αντί για `X`, `OutputPhrase` αντί για `Y`, `TempPhrase` αντί για `Z`, κτλ.

Παραδείγματα στόχων και οι αναμενόμενες απαντήσεις της Prolog:

1. ?- sentence([the, cat, saw, the, dog],[]). Yes
2. ?- sentence([the, saw, cat, the, dog],[]). no
3. ?- sentence(S, []).

S = [the,dog,chased,the,dog] ? ;
 S = [the,dog,chased,the,cat] ? ;
 S = [the,dog,chased,the,boy] ? ;
 S = [the,dog,chased,the,girl] ? ;
 S = [the,dog,chased,a,dog] ? ;
 S = [the,dog,chased,a,cat] ? ;
 S = [the,dog,chased,a,boy] ? ;
 S = [the,dog,chased,a,girl] ? ;
 S = [the,dog,chased,dog] ? ;
 S = [the,dog,chased,cat] ? ;
 S = [the,dog,chased,boy] ? ;
 :

2.5 μονάδες.

Άσκηση 3

Θεωρήσατε την παρακάτω Γραμματική Οριστικών Προτάσεων (ΓΟΠ) «Definite Clause Grammar (DCG)».

S → NP VP
 NP → Det N
 VP → V NP
 VP → V

Det → a, the, ‘τίποτα’

N → dog, cat, boy, girl, dogs, cats, boys, girls

V → chases, chase, sees, see, says, say, believes, believe

Να υλοποιήσετε την παραπάνω γραμματική χρησιμοποιώντας τους DCG κανόνες γραμματικής της Prolog. Θα πρέπει ο μεταγλωτιστής (parser) να λαμβάνει υπόψη του τον αριθμό του υποκειμένου και του ρήματος. Ο αριθμός του αντικειμένου δεν επηρεάζει την σύνταξη της πρότασης. Θα πρέπει ο μεταγλωτιστής σας να αναγνωρίζει την πρόταση «The dog chases cats.» ως συντακτικά σωστή ενώ την πρόταση «The dog chase cats.» θα πρέπει να την απορρίπτει.

Παραδείγματα στόχων και οι αναμενόμενες απαντήσεις της Prolog:

1. ?- sentence([the,dog, chases,cats],[]).
yes
2. ?- sentence([the,dog, chase, cats],[]).
no
3. ?- sentence(S, []).
S = [a,dog,chases,a,dog] ? ;
S = [a,dog,chases,a,cat] ? ;
S = [a,dog,chases,a,boy] ?
:

Σημείωση: Τον αριθμό να τον χειριστείτε ως ένα όρισμα των μη-τελικών συμβόλων. Για παράδειγμα, θεωρήσατε τον παρακάτω κανόνα.

sentence --> noun_phrase(N), verb_phrase(N).

Αυτός ο κανόνας μας λέει ότι μια πρόταση «sentence» αποτελείται από ονοματική φράση «noun_phrase(N)» με κάποιο αριθμό ως χαρακτηριστικό «N», ακολουθείται από μια ρηματική φράση «verb_phrase(N)» με τον ίδιο αριθμό «N».

3.0 μονάδες

Άσκηση 4

Θεωρήσατε την παρακάτω Γραμματική Οριστικών Προτάσεων (ΓΟΠ) «Definite Clause Grammar (DCG)».

S → NP VP

NP → Det N

VP → V NP

VP → V

Det → a, the, ‘τίποτα’

N → dog, cat, boy, girl

V → chase, see, say, believe

Να υλοποιήσετε την παραπάνω γραμματική χρησιμοποιώντας τους ΓΟΠ/DCG κανόνες γραμματικής της Prolog. Θα πρέπει ο μεταγλωττιστής (parser) να λαμβάνει υπόψη του τον αριθμό του υποκειμένου και του ρήματος. Δηλαδή, το υποκείμενο και το ρήμα θα πρέπει να είναι στο ίδιο πρόσωπο. Ο αριθμός του αντικειμένου δεν επηρεάζει την σύνταξη της πρότασης. Θα πρέπει ο μεταγλωττιστής σας να αναγνωρίζει την πρόταση «The dog chases cats.» ως συντακτικά σωστή ενώ την πρόταση «The dog chase cats.» θα πρέπει να την απορρίπτει. Επιπλέον, θα πρέπει για κάθε ουσιαστικό και ρήμα να υπάρχει μόνο στη βασική του μορφή. Δηλαδή, τα ουσιαστικά σε μορφή του ενικού, π.χ. cat όχι cats, ενώ τα ρήματα σε μορφή πληθυντικού (απαρεμφάτου – indefinite), π.χ. chase όχι chases. Η μορφή του πληθυντικού για τα ουσιαστικά θα πρέπει να δημιουργείται από την αντίστοιχη βασική μορφή του ενικού εκτός αν είναι ανώμαλο οπότε θα πρέπει να κρατείται χωριστά στην βάση γνώσης και η μορφή του πληθυντικού. Η μορφή του 3^{ου} προσώπου ενικού για τα ρήματα θα πρέπει να δημιουργείται από την αντίστοιχη μορφή του πληθυντικού (απαρεμφάτου– indefinite).

Σημείωση: Στην DCG γραμματική που θα χρησιμοποιήσετε να παρεμβάλετε κώδικα της Prolog ο οποίος θα προσδιορίζει/ φτιάχνει την μορφή στον πληθυντικό για το ουσιαστικό και στον ενικό για το ρήμα. Τον αριθμό, ενικό ή πληθυντικό, να τον χειριστείτε ως ένα όρισμα των μη-τελικών συμβόλων όπως στην άσκηση 3 αυτής της εργασίας. Για παράδειγμα, όπως στον κανόνα.

sentence --> noun_phrase(N), verb_phrase(N).

Παραδείγματα στόχων και οι αναμενόμενες απαντήσεις της Prolog:

1. ?- sentence([the,dog, chases,cats],[]).

yes

2. ?- sentence([the,dog, chase, cats],[]).

no

3.5 μονάδες