**EXERCITII PARSING TOP-DOWN SI BOTTOM-UP**

**FARA HARTA**

1. Revedeti parser-ul top-down descris si, prin adaugarea de argumente, faceti ca acest parser sa implementeze acordul in numar dintre verb si subiectul sau. Realizati o reprezentare a arborelui de derivare corespunzator.
2. Modificati acelasi parser astfel incat el sa realizeze intotdeauna o reprezentare a arborelui de derivare (chiar si atunci cand regulile sunt date in forma lor initiala, adica fara argumente).
3. Revedeti parser-ul bottom-up cu deplasare-reducere. Adaugati caracteristici acestui parser pentru a se realiza acordul in numar dintre verb si subiectul sau si pentru a se construi o reprezentare a arborelui de derivare corespunzator.
4. Revedeti parser-ul care realizeaza analiza sintactica din coltul stang. Faceti ca acest parser sa lucreze utilizand urmatoarea gramatica:

S NP VP

NP Det N

NP NP Conj NP

VP V NP (PP) (PP) este grup prepositional optional

PP P NP

Det

Det the, all, every

P near

Conj and

N dog, dogs, cat, cats, elephant, elephants

V chase, chases, see, sees, amuse, amuses

(din care se exclude regula D )

1. Adaugati argumente nodurilor regulilor astfel incat acest ultim parser sa construiasca o reprezentare a arborelui de derivare si sa implementeze acordul in numar dintre verb si subiectul sau.
2. Modificati acest parser astfel incat el sa construiasca o reprezentare a arborelui de derivare (fara a necesita argumente ale regulilor).
3. In cazul aceluiasi parser, modificati predicatul completeaza astfel incat acesta sa poata trata constituentii nuli in acea portiune a arborelui pe care o trateaza in maniera

top-down. Apoi modificati gramatica, astfel incat singura regula care rescrie un NP sa fie

NP Det N PP,

unde PP . In cadrul parser-ului modificat, *regula(pp,[ ])* trebuie sa permita parser-ului sa sara peste constituentul PP fara a incerca sa il analizeze.

1. Cum actioneaza perser-ul de tip ,,coltul stang” atunci cand i se cere sa ,,genereze o propozitie” in urma unei interogari de tipul:

?- parse (s, What, [ ] ).

(Obs: O astfel de interogare genera intr-adevar o propozitie in cazul parser-ului

top-down).

1. Se da urmatoarea gramatica:

S NP VP

NP Det N

NP NP Conj NP (a nu se folosi la un parser top-down)

VP V NP (PP) (categoriile puse intre paranteze sunt optionale)

PP P NP (P propozitie)

Det (a nu se folosi la un parser de tip bottom-up)

Det the, all, every

P near

Conj and

N dog, dogs, cat, cats, elephant, elephants

V chase, chases, see, sees, amuse amuses

Construiti tabela cu legaturi pentru intreaga gramatica.

Adaugati tabela de legaturi parser-ului care face analiza sintactica din coltul stang si aratati ca intreaga gramatica poate fi acum folosita in analiza sintactica.

1. Definiti un predicat

generate\_links

care ,,se uita” la regulile unei gramatici, genereaza automat legaturile corespunzatoare si le adauga pe acestea (,,assert”) in memorie.

1. Adaugati reguli parser-ului de tip BUP, reguli care sa acopere in intregime gramatica de la exercitiul nr.9, cu exceptia regulii Det . Aratati ca acest parser functioneaza.
2. In parser-ul din exercitiul precedent adaugati nodurilor argumente care sa realizeze acordul in numar dintr subiect si verb si, in acelasi timp, realizati o reprezentare a arborelui PS in timpul procesului de analiza sintactica.
3. Adaugati legaturi parser-ului BUP astfel incat acesta sa functioneze.