Algorithm:

1. For each nonterminal :

a. Repeat until an iteration leaves the grammar unchanged:

b. For each rule , being a sequence of terminals and non terminals.

2. If begins with a nonterminal and :

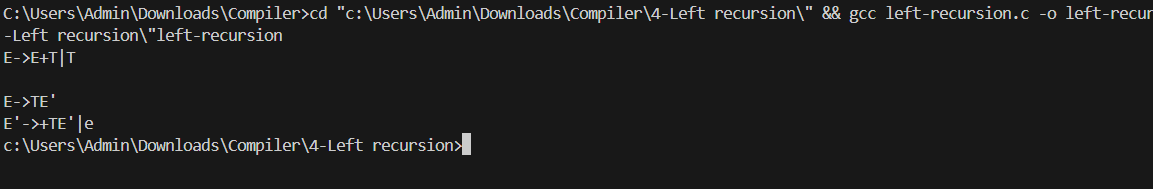
a. Let be without its leading.

b. Remove the rule .

c. For each rule :

d. Add the rule .

3. Remove direct left recursion for as described above.



Algorithm:

1. For each non terminal A find the longest prefix a common to two or more of its alternatives.

2. If a!= E,i. e., there is a non trivial common prefix, replace all the A productions.

3. A-> aβ1| aβ2| .................. | aβn| ◌ where ◌ represents all alternatives that do not begin with a by

4. A==> a A'| ◌

5. A'==>β1|β2| ................. |βn

6. Here A9 is new non terminal. Repeatedly apply this transformation until no two alternatives for a

non-terminal have a common prefix.

