Theory

Find first ( ) in compiler design :-

→ First (A) Contains all terminals present in first place of every string derived by A.

(1) S → abc | def | ghi

(2) first (terminal) = terminal

(3) First (ε) = ε

Find follow ( ) in compiler design :-

→ follow (A) contains set off all terminals present immediate in right of 'A'.

(1) follow of start symbol is $

fo(A) = {$}

(2) S → ACD
C → a/b

fo(A) = first (C) = {a, b}

fo (D) = follows (S) = {$}

(3) S → aSbS | bSaS | ε

follow never contain ε

- **kernel items**; Includes initial item S' --> .S and all items in which dot does not appear at the left most position.

- **Non-kernel items**; All other items which have dots at the leftmost position.

- **Back Patching** :—— Back Patching is basically a process of fulfilling unspecified information.

- **parser**; In computer technology, a parser is a program that's usually part of a compiler.

- **Predictive parser**; A predictive parser is a recursive descent with no backtracking or backup.

- **Dead code elimination**; It is an optimization that removes code which does not affect the program results.

- **Compiler time evaluation**; is the ability of a compiler, that would normally compile a function to machine code and execute it at run time, to execute the function at compile time.

- Variable propagation: It can be defined as the process of replacing the constant value of variables in the expression.

- Induction variable and strength reduction: It lets us "reduce" multiplication operations on IVs to addition operations.

- Code motion: Is a compiler optimization which performs this movement automatically.