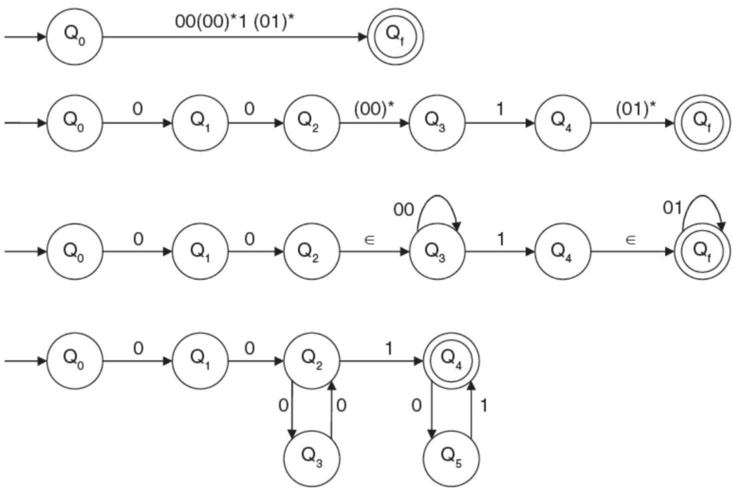
ii) A → (00)*00 S → (01)*S/1(00)*00

S → (01)*1(00)*00

The regular expression is (01)*1(00)*00.
iii) Reversing the regular expression, we get 00(00)*1(01)*.
iv) The fi nite automata constructed from the expression in step III is



v) The right linear grammar from the finite automata is

A → 0B
B → 0C
C → 0D
D → 0C
C → 0E/1
E → 0F
F → 1E/1

This is the equivalent right linear grammar.

6.7 Normal Form

For a grammar, the RHS of a production can be any string of variables and terminals, i.e. , $(VN \cup \sum)^*$. A grammar is said to be in normal form when every production of the grammar has some specifi c form . That means , instead of allowing any member of $(VN \cup \sum)$ on the RHS of the production, we permit only specifi c members on the RHS of the production . But these restrictions should not hamper the language generating power of the grammar.

When a grammar is made in normal form , every production of the grammar is converted in some specifi c form. These help us to design some algorithm to answer certain questions, such as if a CFG