Question:

Give unambiguous CFGs for the following languages.

a. {w| in every prefix of w the number of a’s is at least the number of b’s}

b. {w| the number of a’s and the number of b’s in w are equal}

c. {w| the number of a’s is at least the number of b’s in w}

Answer:

----SETP1----

First step is finding a grammar for finding an unambiguous CFG whether it is ambiguous or unambiguous.

• Grammar for the given problem is

$$S \to SS|(S)|\varepsilon$$
$$S \to a \mid b$$

• But the given grammar is ambiguous as there is more than one parse tree is possible for the above grammar.

• After finding the grammar finds that whether it is ambiguous or not.

• If the grammar is ambiguous then set some priority rules for selecting derivation tree for the grammar and after that set production rules for the grammar.

• Now after setting some priority rules the grammar can be:

$$S \to (S)S|\varepsilon$$
$$S \to a \mid b$$

• But this is also wrong grammar as in this grammar for every prefix of any input string w the number of a’s does not at least the number of b’s.

• Final grammar in which for every prefix of any input string *w* the number of a’s is at least the number of b’s is as shown:

$$S \to (S)S|(S|\varepsilon$$
$$S \to a \mid b$$

----SETP2----

a)

The input alphabet *a* is prefixes at least the number of b’s that means if the length of the input string is one then that must be a.

For the two or more inputs the possible strings are *aa*, *ab*, *aaa*, *aab* and so on.

So, the unambiguous grammar (CFG) is given below:

$$S \to aS \mid a \mid b \mid \varepsilon$$

The production for a regular expression *aaab*.

$$\begin{aligned} S &\rightarrow aS \\ &\rightarrow aaS \\ &\rightarrow aaaS \\ &\rightarrow aaab \end{aligned}$$

This grammar is an unambiguous grammar because there is no possible way to get same regular expression.

----SETP3----

b)

Consider the following unambiguous CFG:

$$S \rightarrow aSb \,|\, bSa \mid \varepsilon$$

In the above grammar, production rules $S \rightarrow aSb$ and $S \rightarrow bSa$ generates the equal number of terminals a’s and b’s.

$$\begin{aligned} S &\rightarrow aSb \\ &\rightarrow a(aSb)b \\ &\rightarrow aa(aSb)bb \\ &\rightarrow aaabbb \end{aligned}$$

----SETP4----

c)

In the given question, the number *a’s* is at least the number of *b’s* that means for a single character input *a* must be the input alphabet.

Consider the following unambiguous CFG:

$$S \rightarrow aSb \,|\, bSa \,|\, aS \,|\, \varepsilon$$

• In the above grammar, production rules $S \rightarrow aSb$ and $S \rightarrow bSa$ generates the equal number of terminals *a’s* and *b’s*.

• The production rule $S \rightarrow aS$ is used for generating the terminal *a’s* as many as user wants.

• The production rule $S \rightarrow \varepsilon$ is used for generating the equal number of terminals *a’s* and *b’s*.

$$\begin{aligned} S &\rightarrow aSb \,|\, bSa \,|\, aS \,|\, \varepsilon \\ &\rightarrow a(aSb)b \\ &\rightarrow aa(bSa)bb \\ &\rightarrow aababb \end{aligned}$$