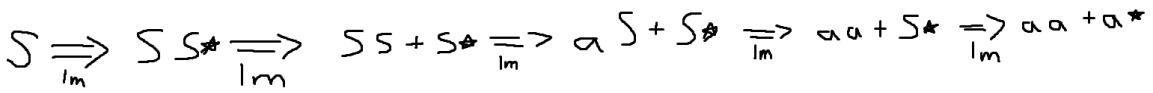
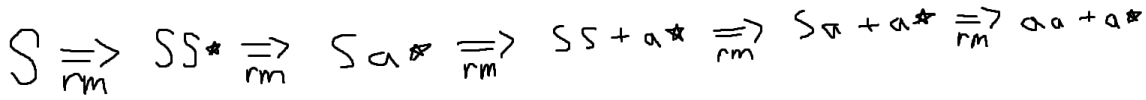
1.

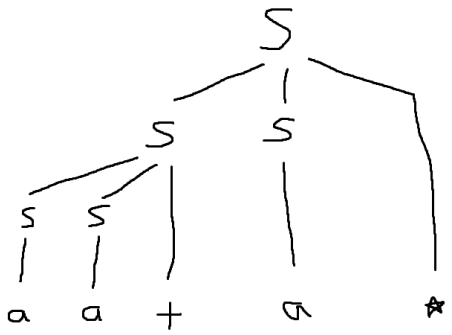
a.



b.



c.



2.

a. S -> A | B
   A -> 1A | 1B | ∈
   B -> 0A

b. S -> 1 S 1 | 0 S 0 | ∈

3.

a. This grammar is already left factored
b. The grammar is not suitable for top down parsing because it has left-recursion.
c. rexpr -> rterm rexpr'
   rexpr' -> + rterm rexpr' | ∈
   rterm -> rfactor rterm'
   rterm' -> rfactor rterm' | ∈
   rfactor -> rprimary rfactor'
   rfactor' -> * rfactor' | ∈
   rprimary -> a | b
d. The Grammar is now suitable for top down parsing because it is left factored and doesn't have left recursion.

4. Remove Left Recursion:

   S -> (S) S' | a S'
   S' -> + S S' | S S' | * S' | ∈

   First(S) = {(, a}
   First(S') = {+, *, ∈, {, a}

Follow(S) = {$, +, *, (, a,)}

Follow(S') = {$, +, *, (, a,)}

Parsing Table:

	(	)	+	*	a	$
S	S→(S)S'				S→aS'	
S'	S'→ϵ S'→SS'	S'→ϵ	S'→ϵ S'→+SS'	S'→ϵ S'→*S'	S'→ϵ S'→SS'	S'→ ϵ