1. **Using an example or illustration, explain what is predictive parsing?**
   It is a type of recursive descent with no backtracking

   Example

   A= BA’

   A’= +BA’|ε

   B= B*C|C

   B’= *CB’|ε

   B= CB’

   C= (A)|k

	First	Follow
A	{(,k}	{$,)}
A’	{+, ε}	{$,)}
B	{(,k}	{),+,$}
B’	{*, ε}	{+,$}
C	{(,k}	{*,+,$,)}

   LL (1) Table

	+	*	(	)	k	$
A			A=BA’		A=BA’	
A’	A’=BA’			A’= ε		A’=ε
B			B=CB’		B=FB’	
B’	B’= ε	B’=*CB’		B’=ε		B’=ε
C			C=(A)		F=k	

2. **a) What is LL (1) Parsing?**
   It is a top-down parsing method in the syntax analysis phase of compiler design. Required components for LL (1) parsing are input string, a stack, parsing table for given grammar, and parser.

   b) Why do we get the First () and Follow () i.e., what do these two functions Really Represent?

   It can be used to prove the LL (K) characteristic of grammar.

   It can be used to promote in the construction of predictive parsing tables.

   It provides selection information for recursive descent parsers

3. **Are there any unique attributes associated with Top-Down parsing?**
   The top-down parsing first creates the root node of the parse tree. And it continues creating its leaf nodes.