**SET A**

1. Draw the transition diagram to recognize numbers in C programming langyage. ( 2 marks)
2. Calculate the first and follow for the given grammar- (3 marks)

SBb/Cd BaB/Ɛ CcC/Ɛ

1. Draw DFA that accept the strings over {a,b} starting with ‘aaa’. (1 mark)
2. Explain shift reduce parser ( 2 marks)
3. Rewrite the grammar by removing left recursion (1 marks)

A🡪 A + B | B

1. Explain S attributed definition. Give example . (2 marks)
2. With explain dead code elimination. (2 marks)
3. Draw the DAG for the following expression. (2 marks)

e = (a\*b) + (c-d) \* (a\*b)

**SET B**

1. Draw the transition to recognize identifiers in C programming languages (2 marks)
2. Calculate first and follow for the following grammar (3 marks)

S → A

A → aB / Ad

B → b

C → g

1. Draw DFA that accepts the string over {a,b} ending with ‘bbb’ (1 mark)
2. Explain LR parser ( 2 marks)
3. Rewrite the grammar by removing left recursion (1 mark)

Z🡪 Z \* Y | Y

1. Explain L attributed definition. Give example (2 marks)
2. With example common subexpression elimination (2 marks)
3. Give the triple of following code (2 marks)

a = b \* - c + b \* - c