OR   iii.   Write down three address code for the following code segment.   **7**

```
if ( x == 1 && y ==2)
{
for (i = 1; i<= n; i++)
{
a = a+b*d-c/f;
}
}
else
{
  while (a >1 )
  a++ ;
}
```

Q.6   Attempt any two:

i.   What is activation record? Explain different storage allocation techniques.   **5**

ii.   Explain call by name and call by need parameter passing techniques with the help of example.   **5**

iii.   Explain different types of code optimization techniques.   **5**

******

Enrollment No......................................

Faculty of Engineering
End Sem (Even) Examination May-2019
CS3CO14 Compiler Design
Programme: B.Tech.   Branch/Specialisation: CSE

Duration: 3 Hrs.   **Maximum Marks: 60**

Note: All questions are compulsory. Internal choices, if any, are indicated. Answers of Q.1 (MCQs) should be written in full instead of only a, b, c or d.

Q.1   i.   Which one of the following languages over the alphabet $\{0,1\}$ is described by the regular expression: $(0+1)*00(0+1)*$?   **1**
     (a) The set of all strings containing the substring 00.
     (b) The set of all strings containing at most two 0's.
     (c) The set of all strings containing at least two 0's.
     (d) The set of all strings that begin and end with either 0 or 1.

ii.   Let L denotes the language generated by the grammar S -> aSa/aa. Which of the following is true?   **1**
     (a) L = a+
     (b) L is regular but not a+
     (c) L is context free but not regular
     (d) L is not context free

iii.   The number of tokens in the following C statement is   **1**
     printf("i = %d, &i = %x", i, &i);
     (a) 10   (b) 31   (c) 18   (d) None of these

iv.   Which of the following statement is/are true?   **1**
     S1: A token can be a keyword.
     S2: In a compiler, keywords of a language are recognized during parsing of the program.
     (a) S1 only   (b) S2 only
     (c) Both (a) and (b)   (d) None of these

v.   Consider the following statements   **1**
     P : Every SLR(1) is LALR(1)
     Q: Every LL(0) is LL(1).
     R: Every LL(1) is LR (1).
     (a) P and Q only   (b) P and R only
   (c) Q and R only   (d) All of these

vi.   Assume that the SLR parser for a grammar G has n1 states and the **1**
       LALR parser for G has n2 states and CLR parser for G has n3 states.
       The relationship between n1, n2 and n3 is:
       (a) n1<n2<n3
       (b) n1 = n2 <= n3
       (c) n1 = n2 < n3
       (d) n1 =n2 = n3.

vii.   Consider the following Syntax Directed Translation Scheme (SDTS), **1**
       with non-terminals {S, A} and terminals {a, b}}.
           S → bA { print "0"}
           S → b { print "1" }
           A → Sa { print "2"}
       Using the above SDTS, the output printed by a bottom-up parser, for
       the input bba is
       (a) 012   (b) 120   (c) 210   (d) 102

viii. In a bottom-up evaluation of a syntax directed definition, inherited **1**
       attributes can
       (a) Always be evaluated
       (b) Be evaluated only if the definition is L attributed
       (c) Be evaluated only if the definition has synthesized attributes
       (d) Never be evaluated

ix.   Which one of the following is/are TRUE?   **1**
       S1: A basic block is a sequence of instructions where control enters
       the sequence at the beginning and exits at the end.
       S2: Available expression analysis can be used for common sub
       expression elimination.
       (a) S1 only   (b) S2 only
       (c) Both (a) and (b)   (d) None of these

x.   Which of the following statement about peep-hole optimization is true? **1**
       (a) It is applied to small part of the code and applied repeatedly
       (b) It can be used to optimize intermediate code
       (c) It can be applied to a portion of the code that is not contiguous
       (d) It is applied in symbol table to optimize the memory requirements.

Q.2   i.   Explain Chomsky hierarchy of language.   **4**
       ii.   Design Minimized DFA that accepts set of all valid binary strings **6**
              whose decimal equivalent is divisible by 64.

OR   iii.   Write down Regular expression for the following DFA (using Arden's **6**
             theorem)
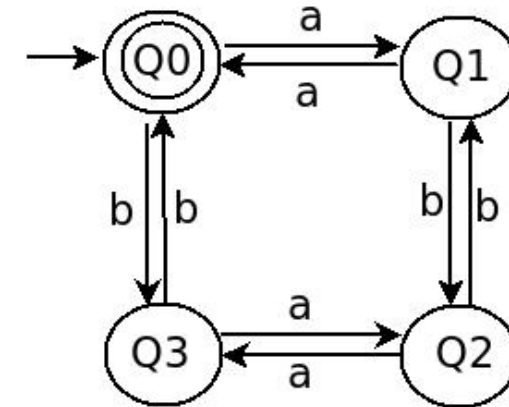
Q.3   i.   What is token?   **2**
       ii.   What is LEX? Explain auxiliary definition of LEX tool with the help **8**
              of simple program.

OR   iii.   (a) Explain different phases of compiler with the help of example. **8**
             (b) Explain bootstrapping, cross complier and input buffering.

Q.4   i.   Write down difference between top down and bottom up parsing.   **3**
       ii.   Check the following grammar is LL(1) or not with the help of **7**
              predictive parsing table. Note: Ɛ represents empty string.
              S → ABC
              A → aA / a
              B → bB / b
              C → Ce / Ɛ

OR   iii.   Design LALR (1) table for the following grammar and then check it is **7**
             LALR (1) or not? Note: Ɛ represents empty string.
             S → AaBb
             A→ aAB / a
             B → AB / Ɛ

Q.5   i.   Explain Dependency Graph with the help of example.   **3**
       ii.   What is attributed Grammar? Explain Synthesised attributed grammar **7**
              and inherited attributed grammar with the help of example and also
              explain relationship between Synthesised attributed grammar and
              inherited attributed grammar.

Marking Scheme
CS3CO14 Compiler Design

Q.1　i.　Which one of the following languages over the alphabet {0,1} is described by the regular expression: (0+1)*00(0+1)*?　**1**
(a) The set of all strings containing the substring 00.

　　ii.　Let L denotes the language generated by the grammar S -> aSa/aa.　**1**
Which of the following is true?
(b) L is regular but not a+

　　iii.　The number of tokens in the following C statement is　**1**
printf("i = %d, &i = %x", i, &i);
(a) 10

　　iv.　Which of the following statement is/are true?　**1**
(a) S1 only

　　v.　Consider the following statements　**1**
P : Every SLR(1) is LALR(1)
Q: Every LL(0) is LL(1).
R: Every LL(1) is LR (1).
(d) All of these

　　vi.　Assume that the SLR parser for a grammar G has n1 states and the LALR parser for G has n2 states and CLR parser for G has n3 states. The relationship between n1, n2 and n3 is:　**1**
(b) n1 = n2 <= n3

　　vii.　Consider the following Syntax Directed Translation Scheme (SDTS), with non-terminals {S, A} and terminals {a, b}}.　**1**
(b) 120

　　viii.　In a bottom-up evaluation of a syntax directed definition, inherited attributes can　**1**
(b) Be evaluated only if the definition is L attributed

　　ix.　Which one of the following is/are TRUE?　**1**
(c) Both (a) and (b)

　　x.　Which of the following statement about peep-hole optimization is true?　**1**
(a) It is applied to small part of the code and applied repeatedly

Q.2　i.　Explain Chomsky hierarchy of language.　**4**
FOUR types of language 1 mark for each　(1 mark * 4)

　　ii.　Valid Minimized DFA　**6**

OR　iii.　Regular expression for the following DFA (using Arden's theorem)　**6**
Stepwise marking (for each valid step 1 mark)
(1 mark * 6)

Q.3　i.　Definition of token　**2**

　　ii.　LEX　2 marks　**8**
Auxiliary definition of LEX tool　3 marks
Simple program.　3 marks

OR　iii.　(a) Different phases of compiler　3 marks　**8**
Example.　1 mark
(b) Bootstrapping　1.5 mark
Cross complier　1.5 mark
Input buffering.　1 mark

Q.4　i.　Any three difference between top down and bottom up parsing.　**3**
1 mark for each　(1 mark * 3)

　　ii.　Design predictive parsing table　5 marks　**7**
For correct result　2 marks

OR　iii.　Stepwise LALR table　6 marks　**7**
For correct result　1 mark

Q.5　i.　Dependency Graph　2 marks　**3**
Example.　1 mark

　　ii.　Attributed Grammar　2 marks　**7**
Synthesised attributed grammar　1 mark
Inherited attributed grammar　1 mark
Example 1 mark for each (1 mark * 2)　2 marks
Relationship between them　1 mark

OR　iii.　Three address code for the following code segment.　**7**
Stepwise marking (0.5 mark for each correct step)
(0.5 mark * 14)

Q.6　　Attempt any two:
　　i.　Activation record　2 marks　**5**
Any three storage allocation techniques.
1 mark for each (1 mark * 3)　3 marks

　　ii.　Call by name　1 mark　**5**
Call by need parameter　1 mark
Examples1.5 mark for each (1.5 mark *2)　3 marks

　　iii.　Types of code optimization techniques　**5**
Any five techniques 1 mark for each　(1 mark * 5)

*****