## CENTRAL INSTITUTE OF TECHNOLOGY KOKRAJHAR

(Deemed to be University) KOKRAJHAR :: BTR :: ASSAM :: 783370

## $\frac{END-SEMESTER\ EXAMINATION}{UG}$

Session: Janu-June, 2023 Semester: VI Time: 3Hrs. Full Marks: 100 Course Code: UCSE601 Course Title: Compiler Design

## Answer question no 1 and any four from the rest

1. Answer all the questions:

- a) Lexical analyser is related to (regular expression, context free grammar, type checking, left-most derivation) ......
- b) The number of token in the C-statement printf("Hello World"); is ......

c) The grammar  $S \rightarrow a/ab/abc$  is [LL (1), LL (2), LL (3)] .....

d) The regular expression for the language  $L = \{ w \in (0, 1) *: |w| \text{ is even} \}$  .....

e) The most powerful parser is (CLR, SLR, LALR) .....

- f) The bottom-up parsing is also known as (Shift-reduce, Predictive, Recursive descent) .....
- g) Write the three address code for x = a[i][j]

h) Give an example of semantic error.

- i) In order to calculate  $x^n$ , the minimum number of multiplication is .....
- j) What are the different ways to express three address codes?

 $2 \times 10$ 

2. (a) Remove left recursion from the following grammar:

$$A \to ABd|Aa|a, B \to Bc|b|$$

(b) Calculate the First and Follow for the given grammar:

$$S \to ACB|CbB|Ba, A \to da|BC, B \to g|\epsilon, C \to h|\epsilon$$

c) Write quadruples for the expression:

$$(a*b) + (c+d) - (a+b+c+d)$$

6 + 10 + 4

3. a) Consider the following grammar:

$$S \to S, S \to SS|a|\epsilon$$

- i) Construct the collection of sets of LR(0) items for this grammar and draw its go to graph.
- ii) Indicate the shift-reduce and reduce-reduce conflict (if any) in the various state of the LR(0) parser.
- b) Define ambiguous grammar. Check whether the grammar is ambiguous or not? Justify your answer.

$$S \rightarrow aBC, \, A \rightarrow bC|cd, \, C \rightarrow cd, B \rightarrow c|d$$

15 + 5

- 4. (a) Find the basic blocks and construct the Flow Graph of the following piece of Three Address Code:
  - 1. location = -1
- 8. goto 10
- 2. i = 0
- 9. location = i
- 3. if i<100 goto 5
- 10. t3 = i+1
- 4. goto 13
- 11. i = t3
- 5. t1 = 4\*i6. t2 = A[t1]
- 12. goto 3
- 13. return
- 7. if t2 == x goto 9
- b) construct the LL(1) parsing table of the following grammar

$$S \to iEtSS'|a, S' \to eS|\epsilon, E \to b$$

10 + 10

5. (a) Consider the following grammar productions and the corresponding semantic rules:

Semantic Rule
E.val = R.val, $R.inh = T.val$
R.val = R.inh
R.val = R.inh + E.val
T.val = S.val, S.inh = F.val
S.val = S.inh
S.val = S.inh * T.val
F.val = n.val
F.val = E.val

Use this to evaluate the expression 3\*5. Display the annotated parse tree and order of evaluation of the variable attributes.

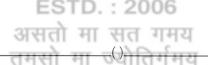
order of evaluation of the variable attributes. (b) The lexical analyzer uses the given patterns for recognizing three tokens,  $T_1$ ,  $T_2$ , and  $T_3$ , over the alphabets  $\{a,b,c\}$ .

 $T_1 : a?(b|c)*a$   $T_2 : b?(a|c)*b$  $T_3 : c?(b|a)*c$ .

Note: 'x?' means 1 or 0 occurrences of the symbol x. Also, the analyzer outputs the token matching the longest possible prefix. If the analyzer processes the string bbaacabc, find the sequence of tokens it outputs.

10 + 10

- 6. Write the short note on (any two)
- a) Peephole optimization
- b) Lex and Yacc
- c) Target Code Generation



10 + 10