

The Assam Kaziranga University

End-Semester Examination: May-June, 2017

Programme: BTech

Semester: 6th

Branch/Stream: CSE

Course Code: ET1348

Course Name: Compiler Design

Time: 3 hours

Total Marks: 70

All the students are instructed to write their ID number in the question paper

Q1. Answer all questions (Each question carries 1 mark):

10x1=10

i) CFG can be recognized by

- a) Push down automata
- b) 2 way linear bounded automata
- c) Turing machine
- d) None of the above

ii) A top down parser generates

- a) Left most derivation
- b) Right most derivation
- c) Right most derivation in reverse
- d) Left most derivation in reverse

iii) A given grammar is said to be ambiguous if

- a) Two or more productions have the same non terminal on the left hand side
- b) A derivation tree has more than one associated sentence
- c) There is a sentence with more than one derivation tree corresponding to it
- d) Parenthesis are not present in the grammar.

iv) Three address code involves

- a) Exactly 3 addresses
- b) At most 3 addresses
- c) No unary operator

d) None of the above.

v) Which data structure is used during shift reduce parsing?

- a) Stack
- b) Queue
- c) Array
- d) Pointer

vi) If x is terminal then $FIRST(x)$ is

- a) λ
- b) x
- c) x^*
- d) None of the above

vii) Which of the following is the most powerful parser?

- a) SLR
- b) LALR
- c) Canonical LR
- d) Operator-Precedence

viii) Choose the correct statement

- a) There are CFG's that are not LR
- b) an ambiguous grammar can never be LR
- c) an ambiguous grammar can be LR
- d) Any CFG has to be LR.

ix) Recursive Descent parsing is an example of

- a) Top-down parsing
- b) Bottom-up parsing
- c) Predictive parsing
- d) One of the above.

- x) YACC builds up
- a) SLR parsing table
 - b) Canonical LR parsing table
 - c) LALR parsing table
 - d) None of the above.

Answer any four (from Q2 to Q7) of the following (Each questions carries 15 marks)

Q2.

- a) Explain the various phases of compiler in detail? Also write the output for the following expression after each phase $a=b*c-d$. 8
- b) Draw NFA for the regular expression $(a/b)^*abb$. Obtain DFA from NFA. 4+3

Q3.

- a) Construct the predictive parsing table for the following CFG

$E \rightarrow TE'$

$E' \rightarrow +TE'$

$T \rightarrow FT'$

$T' \rightarrow *FT'$

$F \rightarrow (E)/id$

and parse the following string $id+id*id$ using the predictive parsing table

7+5

- b) Explain whether the grammar is LL(1) or not.

3

Q4.

- a) Find the language from

$S \rightarrow 0S1/0A1$

$A \rightarrow 1A0/10$

2

- b) Define annotated parse tree. Give one example and describe briefly.

2+4

- c) Remove left recursion of the following grammar

$S \rightarrow Aa/b$

$A \rightarrow Ac/Sd/\lambda$

4

3

d) Define token, pattern, lexims

Q5.

a) Find the DAG of the following expression

$d = b * c$

$e = a + b$

$b = b * c$

$a = e - d$

5

c) Draw the Syntax tree, DAG, Triples, Quadruples and Indirect triples for the expression $a = b * -c + b * -c$

$5 \times 2 = 10$

Q6.

Construct the SLR parsing table for the following CFG

$E \rightarrow E + T / T$

$T \rightarrow T * F / F$

$F \rightarrow (E) / id$

and parse the following string $id * id + id$ using the SLR parsing table.

$8 + 7$

Q7.

Write short note on any five of the following

$3 \times 5 = 15$

a) Peephole optimization

c) Cross compiler

d) Context free grammar

e) DAG

f) LEX

g) Symbol Table