Nirma University

Institute of Technology

Semester End Examination (IR), December - 2019

B. Tech. in Computer Engineering, Semester-VII

IT794 Compiler Construction

Roll / Exam No. | Supervisor's Initial with Date

Time: 3 Hours | Max Marks: 100

Instructions:
1. Attempt all the questions.
2. Figures to right indicate full marks.
3. Draw neat sketches wherever necessary.
4. Assume suitable data wherever required.

Q-1 **Do as directed** **[18]**

A CO-1 BL-2 Define ambiguous grammar. Show that the following grammar is ambiguous or not. [6]

A → AA | (A) | a

B CO-1 BL-2 With a given semantic rule: **(ε is NULL)** [6]

Production	Semantic Rules
T → F T'	T'.inh = F.val T.val = T'.syn
T' → * F T1'	T1'.inh = T'.inh * F.val T'.syn = T1'.syn
T'→ ε	T'.syn = T'.inh
F→ digit	F.val = digit.lexval

Draw annotated parse tree for the given expressions. 8*9

C CO-1 BL-2 Show whether the following grammar is SLR or not. [6]

S → A+S

S → A

A→ a

Q-2 **Do as directed** **[16]**

A CO-2 BL-3 Whether the given grammar is LL(1) or not. Prove your answer with proper parsing table and explanation. (ε is NULL) [8]

S → iEtSS' | a

S' → eS | ε

E → b

OR

A CO-2 BL-3 Show that the following grammar is CLR(1) but not LALR(1). [8]

S->Aa/bAc/Bc/bBa
A->d
B->d

Explain with proper state diagram and parsing table.

B CO-2 BL-3 Show Whether the following grammar is LALR(1) or not. [8]

S → AA
A → aA
A → b

Explain with proper state diagram and parsing table.

Q-3 **Do as directed** **[16]**

A CO-3 BL-4 Define terms Activation Tree and Activation Record in static allocation of space with example. [4]

OR

A CO-3 BL-4 Explain the following code optimization techniques with example. [4]
a. Constant folding
b. Strength Reduction

B CO-3 BL-4 Write down the single static assignment for given below statements. [4]

p = a ^ b
q = p * c
p = q / d
p = e + p
q = p – q

How many variables are required to convert into static single assignment?

C CO-3 BL-4 Construct the DAG for the given below expression [4]
((x + y) - ((x + y) /(x – y))) + ((x + y) * (x – y))

D CO-3 BL-4 Translate the arithmetic expression a*(b-c)+d/e into [4]
1. A syntax tree.
2. Quadruple.

Section II

Q-4 **Do as directed** **[16]**

A CO-1 BL-2 Differential Synthesized attributes and inherent attributes with suitable example. [4]

B CO-1 BL-2 Explain the following issues in the design of a code generator? [4]
1. Register allocation
2. Instruction Selection

C CO-1 BL-2 Convert the following C code into 3 address code: [4]

```
fact(x)
{
    int f = 1;
    for (i = 2; i <= x; i++)
        f = f * i;
    return f;
}
```

D CO-1 BL-2 Differentiate the Data Flow Analysis and Control Flow Analysis with suitable example. [4]

OR

D CO-1 BL-2 Describe the Code motion and Dead code Elimination optimization technique with proper example [4]

Q-5 **[18]**

A CO-2 BL-3 What do you mean by backpatching?.Show the true and false lists for each expression? You may assume the address of the first instruction generated from 100. [6]

a==b|| (c==d && e==f)

B CO-2 BL-3 Copy propagation and Global common subexpressions optimization techniques are useful for optimizing code, Justify with proper example. [6]

C CO-2 BL-3 What is the relation between S-Attributed definition and L-attributed definition? Write down Which of the following definition are S-attributed and L-attributed. [6]

1. P → QR { R.val = Q.val }
2. P → QR { R.val=Q.val and P.val=R.val }
3. P → QR { Q.val=R.val and P.val=R.val }

Q-6 **Do as directed** **[16]**

A CO-3 BL-4 Differentiate between type checking and type casting with suitable example. [8]

OR

A CO-3 BL-4 Explain Global Register Allocation, Usage counts and Graph coloring interference method to allocate registers. [8]

B CO-3 BL-4 True/False with Justification [8]

1. Left recursive grammar can be LL(1).
2. Reduce/Reduce conflicts and Shift/Reduce Conflicts occur at the same time.
3. The languages that need heap allocation in the runtime environment are those that allow dynamic data structure.
4. Normally type checking is done during code optimization.