SVKM's NMIMS

Mukesh Patel School of Technology Management & Engineering

Programme: B. Tech -Computer Year: III Trimester: IX

Academic Year : 2011-2012

Batch: 2010-2011

Subject: **Principles of Compiler Design** Marks: 100

Time: 2.00 pm To 5.00 pm

Date: 28/04/2012 Duration: 3 hrs.

Re - Examination

Instructions: Candidate should read carefully the instructions printed on the question paper and on the cover of the Answer Book, which is provided for their use.

1. Question No-1 is compulsory
2. Out of remaining questions, attempt any 4 questions
3. In all five questions to be attempted
4. All question carry equal marks.
5. Answer to each new question to be started on a fresh page.
6. Figures in bracket on the right hand side indicate full marks.

1. Attempt All **[2*10 =20]**

a) What is a translator? Discuss the role of various phases of the compiler in the translator of source program to object code.

b) Explain cross compiler. Suppose you have a working C compiler on machine A. Discuss the steps you would take to create a working compiler for another language C' on a machine B.

2. Attempt All **[2*10 = 20]**

a) Discuss the action taken by every phase of the compiler on the knowing string:

$$A = B * C + D / E$$

b) What is the use of deterministic finite automata in lexical analysis? Also give example

3. Attempt All **[2*10 =20]**

a) Consider the following grammar:

G:

$$E \rightarrow E * T / T$$

$$T \rightarrow T * F / F$$

$$F \rightarrow (E) / id$$

Construct the LR (0) canonical collection and also design the SLR parsing table.

b) Give Algorithm for construction of predictive parsing table. Consider the following grammar and construct predictive parsing table:

$$S \rightarrow iEtSS_1$$
$$S_1 \rightarrow eS / E$$
$$E \rightarrow b$$

4. Attempt All **[2*10 = 20]**

a) Consider the following grammar:

$$E \rightarrow E+T$$
$$E \rightarrow T$$
$$T \rightarrow T*F$$
$$T \rightarrow F$$
$$F \rightarrow id$$

And write the intermediate code for expression `a + b * c` as:

i) Postfix representation
ii) Syntax tree representation
iii) Three address code representation

b) Explain the back patching? Write the semantic action for following productions:

S→ `begins L end`
$S \rightarrow A$
$L \rightarrow L_1; MS$
$L \rightarrow S$

5. Attempt All **[2*10 = 20]**

a) What do you mean by DAG? Explain the algorithm for constructing a DAG with the help of suitable example.

b) What is LR parser? How it is different from SLR. Construct LALR table for:

$$S \rightarrow Ba / bBc / dc / bda$$
$$B \rightarrow d$$

6. Attempt All **[2*10 = 20]**

a) For the grammar having the productions:

$$A \rightarrow (A)A / \varepsilon$$

Compute FIRST and FOLLOW set of A.

b) Consider the following grammar:

$$E \rightarrow E + E$$
$$E \rightarrow E + E$$
$$E \rightarrow (E)$$
$$E \rightarrow id$$

Using the above grammar, for the input string $id_1 + id_2 * id_3$ show the stack implementation for shift reduce parsing.

7. Attempt any two **[2*10 = 20]**

a) Left recursion and left factoring and how these are eliminated explain with suitable example.

b) Discuss principal sources of optimization.

c) Discuss the role of data flow analysis.