

Seventh semester examination-2006 COMPILER DESIGN

Full marks-70

Time-3hrs

Answer Question No.1 which is compulsory and any five from the rest.

The figure in the right side margin indicates the mark.

1. Answer the following questions: [2x10]
 - a) Explain why it is possible to design an independent Lexical Analyzer?
 - b) Define and differentiate between compile time errors and runtime errors?
 - c) Explain the machine dependant and machine independent code optimization?
 - d) Explain and difference between Bottom-up and Top-down parsing?
 - e) What are the drawbacks of SLR(1) parser?
 - f) What do you mean by porting of a compiler?
 - g) Describe the structure used to create a symbol table?
 - h) Describe various data structures used to create a symbol table?
 - i) Distinguish between syntax and semantics of programming language? Explain which Part of a compiler are permanently concerned with each.
 - j) Which are the major functioning of the five main stages of compiler.

2. a) For the following grammar ,find the FIRST and FOLLOW sets of each of the non terminals

$$S \rightarrow aAB|bA|\epsilon$$

$$A \rightarrow aAb| \epsilon$$

$$B \rightarrow bB|c$$
 - b) Differentiate between syntax directed definition and syntax directed translation scheme.
 - c) Test whether the following grammar LL(1)?

$$s \rightarrow aAB$$

$$A \rightarrow cd|ef$$
 - d) Explain the concept of bootstrapping in compiler design process.

[2.5x4]

3. a) Use T-diagram to describe the steps you would take to create a powerful compiler using a Quick dirty compiler. [2]
- b) Define and discuss the objectives of SDTS? What do you mean by underlying source grammar?

- Explain with example. [4]
c) Construct the DAG for the following statement. [4]

$$Z = X - Y + x * Y * U - V / W + X + V$$

4. a) Describe the contents of a symbol table. How is the Symbol table involved in the interactions

Between the different components of the compiler and in error detection? [5]

Give a simple example in each case.

- b) Explain the machine dependant and machine independent code optimization. What are their their advantages. [5]

5. a) Explain the working principle of operator precedence parsing algorithm. Explain the

| | - | * | / | ↑ | Id | \$ |
|---|----|----|----|----|----|----|
| - | •> | <• | <• | <• | <• | •> |
| * | •> | •> | <• | <• | <• | •> |
| / | •> | •> | •> | <• | <• | •> |
| ↑ | •> | •> | •> | •> | <• | •> |
| ↑ | •> | •> | •> | •> | <• | •> |

Parsing action for the input string $id_1-id_2/id_3*id_4 \uparrow id_5-id_1$ with the reference to the operator Precedence relation table given below. [5]

- b) what information is recorded in the symbol table of a compiler for a block structure language? Give examples of how this information is created and /or used at each stage of compilation. [5]

6. a) Construct an LL(1) parsing table for the following grammar [8]

$$S \rightarrow aBDh$$

$$B \rightarrow cC$$

$$C \rightarrow bC \mid \epsilon$$

$$D \rightarrow EF$$

$$E \rightarrow g \mid \epsilon$$

$$F \rightarrow f \mid \epsilon$$

b) Explain how scope rules and the block structure of a programming language decides the structure of symbol table.

7. a) Construct the SLR parsing table for the following grammar

[8]

$E \rightarrow E+T$

$E \rightarrow T$

$T \rightarrow T * F$

$T \rightarrow F$

$F \rightarrow id$

$L \rightarrow L, E/E$

b) What is the objective of intermediate code generation? What is the different form of intermediate code generated by intermediate code generation phase?

[2]

8. a) What is the objective of intermediate code generation? Generate three address code for the following code segment.

[4]

Main()

{

Int a=1;

Int b[10];

While(a<=10)

 B[a]=2**a;

}

b) Find the canonical collection of sets of LR(1) items

[3]

$S \rightarrow AaAb$

$A \rightarrow BbBa$

$A \rightarrow \epsilon$

$B \rightarrow \epsilon$

c) Write quadruples, triples and indirect triples for the following expression

[3]

$x[i] := y$

$x := y[i]$

