Seventh Semester Examination – 2007

COMPILER DESIGN

Full Marks – 70

Time – 3 Hours
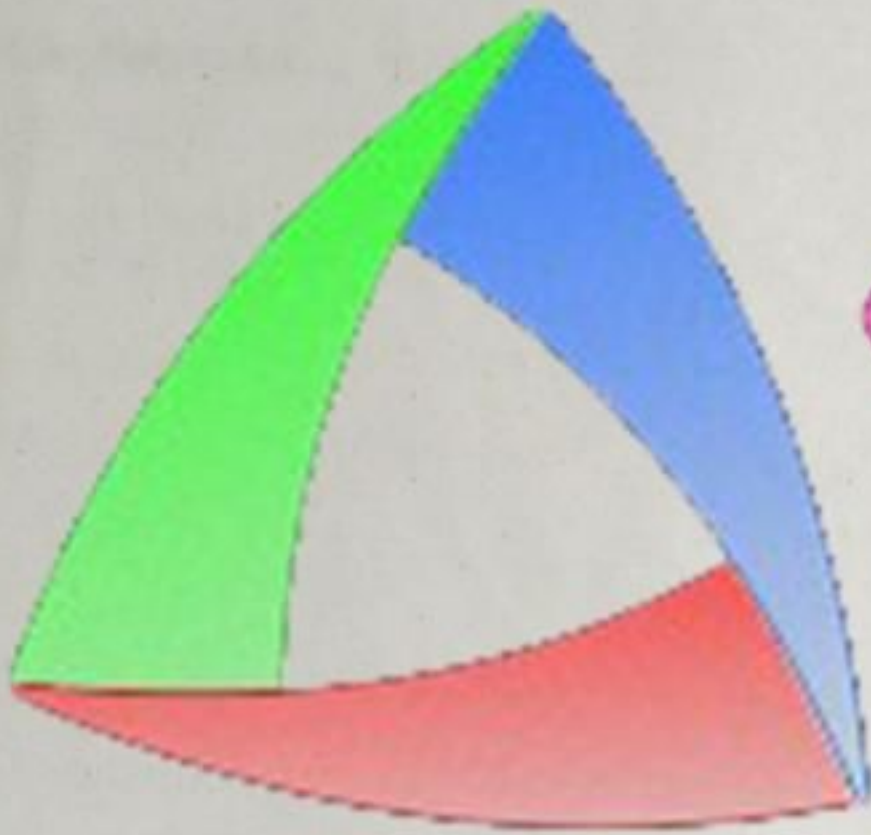
Answer question 1 which is compulsory
and any **five** from the rest.

The figures in the right-hand margin
indicate marks.

1.   Answer the following questions :   2×10

(a) What is syntax directed translation scheme ? What are the different forms of intermediate code used in compilation process ?

(b) What is the *dead code* elimination ?

(c) What is reduce-reduce (R-R) conflict in LR parser ?

(d) Why LR parsing is prefer over other parsers ?

(e) What do you mean by runtime storage allocation ?

(f) Eliminate left-recursion from the following grammar.

$$E \rightarrow aa \mid abba \mid E\,b \mid E\,E$$

(g) Briefly describe one use for flow graphs in compiler writing.

(h) Explain the concept of bootstrapping in compiler design process.

(i) What is back-patching in the process of intermediate code generation ?

(j) Differentiate between phase and a pass in compiler construction.

2. (a) What is the role of intermediate code generation in overall compiler design? 4

(b) Define operator precedence relation and operator precedence grammar. Construct precedence function for the following precedence relation. 6

	−	*	/	↑	id	$
−	•>	<•	<•	<•	<•	•>
*	•>	•>	<•	<•	<•	•>
/	•>	•>	•>	<•	<•	•>
↑	•>	•>	•>	<•	<•	•>
id	•>	•>	•>	•>		•>

3. (a) Discuss the construction of **LR** parser. What are the carious data structures used in **LR** parser design ? Discuss the construction of **ACTION[ ]** and **GOTO[ ]** table. 5

(b) Write the role of an error detector in compilation process ? Discuss different errors in Lexical-Phase. 5

4. (a) What are the necessity of optimization in compilation ? Discuss the factors influencing optimization ? 5

(b) Explain the symbol table construct for the block structure programming language ? 5

5. Consider the following grammar :

$$E \rightarrow (L) \mid a$$
$$L \rightarrow L, E \mid E$$

(a) Construct DFA of LR (0) items for this grammar.

(b) Construct the SLR (1)-parsing table.

(c) Show the parsing stack and actions of an SLR(1) parser for the input string

i. $((a),a,(a,a))$

(d) Is this grammar a LR (0) grammar ? If not describe the LR (0) conflict. 10

6. (a) What is an activation record ? Explain clearly the components of an activation record. 5

(b) Construct DAG for the following sequence of statements

$$X = Y/Z$$
$$W = P^* Y$$
$$Y = Y ^*Z$$
$$P = W - X$$

Perform code generation assuming only one registers is available. 5

7. (a) Consider the following context-free grammar. Where S is the start symbol, and the terminals are a , ( )

$$S \rightarrow ()$$
$$S \rightarrow a$$
$$S \rightarrow (A)$$
$$A \rightarrow S$$
$$A \rightarrow A, S$$

5

P.T.O.

Show precisely why this grammar is not LL(1). Rewrite this grammar to make it suitable for recursive descent parsing. 5

(b) Discuss the importance of symbol table in compiler design. How is the symbol table manipulated at various phase of compilation ? 5

8. (a) Find the FIRST and FOLLOW sets for each of the non-terminals in the following grammar. (in the grammar below ∈ denotes epsilon, the empty string) 4

A → aBa

B → bCb | bcD

C → cCc | ∈

D → Deb | ∈

(b) Differentiate between syntax directed definition and syntax directed translation scheme. 4

(c) Explain, why it is possible to design an independent Lexical analyzer. 2