B. Tech

PECS 3404

Seventh Semester Examination – 2008

COMPILER DESIGN

Full Marks – 70

*Answer Question No. **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 the front-end of a compiler ?

   (b) What are the error recovery actions in a lexical analyzer ?

   (c) What are the goals of error handler in a parser ?

(d) What do you mean postfix translations ?

(e) Explain, why a common approach are being used to design the Lexical analyzer for different compilers ?

(f) What is basic block in a code segment ?

(g) Why do most compilers contain symbols tables ?

(h) What is the need of attributed grammar ar L-attributed grammar in semantic analysis ?

(i) Suggest a efficient data structure to implement LR parsing table.

(j) Define token and lexeme. Find out the tokens from the following code segment.

```
main()
{
int    a = 5;
int    b[11];
while (a <= 5)
        b[a] = 3 * a;
}
```


2. (a) Describe the basic differences between top-down and bottom-up parsing. If you had to write a parser by hand which approach would you use ? If you had to write a parser for a relatively complex language, which approach would you use ? 5

(b) Show that the following is an SLR (1) grammar. 5

$e \rightarrow e\ ADD\ t \mid t$

$t \rightarrow t\ MUL\ f \mid f$

$f \rightarrow VAR$

3. (a) What is the objective of intermediate code generation ? Write quadruples, triples and indirect triples for the following expression. 5

$-(a + b) * (c + d) - (a + b + c)$

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

4. (a) Explain the difference between LALR parsing table and canonical LR parsing table ? 5

(b) Write the algorithm to construct a canonical LR parsing table for the augment grammar. 5

5. (a) Consider the following context-free grammar. Where S is the start symbol.

$S \rightarrow A\ S \mid b$

$S \rightarrow (\ A\ )$

$A \rightarrow a$

$A \rightarrow S\ A$

Construct an NFA whose sets are the LR(0) items of the given grammar. Show that the canonical collection of LR(0) items for the grammar is the same as the states of equivalent DFA. 5

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

6. (a) What are the issues in the design of the code generator ? Explain. 5

(b) What is DAG ? Write an algorithm to construct DAG from the block of three address code. Explain the process with a relevant example. 5

7. (a) What is a three address code ? What are its types ? How it is implemented ? Generator three address code for the following code segment. 5

```
main()
{
int     a = 1;
int     b[10];
while (a <= 10)
        b[a] = 2 *a;
```

(b) Discuss the structure of a symbol table. Explain how the symbol table is created for a block structured language. 5

8. (a) Write down the necessary algorithms and define FIRST and FOLLOW. 5

(b) Let the Target machine is a byte addressable machine with four bytes to a word and n general-purpose registers, $R_0, R_1, \ldots, R_{n-1}$. It operates on two address instruction of the form op source, destination; The op-codes are MOV, ADD, SUB.

Generate code for the following C statements for a target machine described above. Assume that three registers are available. 5

(i) $x = f(a) + f(a) + f(a)$

(ii) $a[i] = b[j]$

(iii) $a[i] = b[c[i]]$

(iv) $a[i] = a[i] + b[j]$.