(Following Paper ID and Roll No. to be filled in your Answer Book)

PAPER ID : 1038 Roll No. □□□□□□□□□□

B.Tech.

SIXTH SEMESTER EXAMINATION, 2005-2006

COMPILER CONSTRUCTION

Time : 3 Hours *Total Marks : 100*

Note : *(i)* *Answer* ***ALL*** *questions.*

(ii) *All questions carry equal marks.*

(iii) *In case of numerical problems assume data wherever not provided.*

(iv) *Write your answers as concisely as possible, consistent with providing a complete answer to the question.*

(v) *Be precise in your answer.*

1. Attempt *any four* parts of the following : **(5x4=20)**

(a) Explain bootstrapping in detail.

(b) Given the C assignment

a[i+1] = a[i]+2

draw a parse tree and syntax tree for the expression.

(c) Suppose, you have a Pascal to C translator written in C and a working C compiler. Use T diagram to describe the steps you would take to create a working Pascal Compiler.

(d) Construct a deterministic finite automata (DFA) for the regular expression.

$a.(a+b)^*.b.b$

(e) Write English description for the language generated by the following regular expressions :

(i) $(a \mid b)^*a(a \mid b \mid c)$

(ii) $(aa \mid b)^* \ (a \mid bb)^*$

(f) Write a context sensitive grammar that generates strings of the form xcx, where x is a string of a's and b's.

2. Attempt *any two* parts of the following : **(10x2=20)**

(a) Construct an LALR(1) parsing table for the following grammar :

$D \rightarrow L{:}T$

$L \rightarrow L, id/id$

$T \rightarrow integer$

(b) Consider the following grammar :

$E \rightarrow EBE$

$E \rightarrow num$

$E \rightarrow (E)$

$B \rightarrow +$

$B \rightarrow -$

$B \rightarrow *$

$B \rightarrow \backslash$

(i) Explain why this grammar is not suitable to form the basis for a recursive descent parser.

(ii) Use left-factoring and left -recursion removal to obtain an equivalent grammar which can be used as the basis for a recursive descent parser.

(c) Explain shift reduce parsing and operator precedence parsing.

3. Attempt *any two* parts of the following : **(10x2=20)**

(a) Consider the following code fragment generate three address code for it.

for (i=1; i <=10; C++)

if a < b then $x=y+z$.

(b) Activations of functions within a program are allocated a stack frame. What is the layout of a typical stack frame on the stack ? Identify the kinds of data values which are stored there.

(c) Discuss in details about Syntax directed translation schemes, parse tree and syntax trees.

4. Attempt *any two* parts of the following : **(10x2=20)**

(a) What are several methods of organizing the symbol table. Explain.

(b) Write the syntax directed translations to go along with the LR parser for the following :

$$\mathrm{L} \rightarrow \mathrm{id\ elist}$$

$$\mathrm{elist} \rightarrow \mathrm{elist}\ [\mathrm{E}]\ |\ [\mathrm{E}]$$

$$\mathrm{E} \rightarrow \mathrm{E+T}\ |\ \mathrm{T}$$

$$\mathrm{T} \rightarrow \mathrm{T*F}\ |\ \mathrm{F}$$

$$\mathrm{F} \rightarrow \mathrm{id}$$

(c) There are syntactic errors in the following constructs : For each of these constructs, find out which of the input's next tokens will be detected as an error by the LR parser.

(i) While a=b do $x=y+z$

(ii) a+b=c

(iii) a*+b+c

5. Attempt *any two* parts of the following : **(10x2=20)**

(a) (i) Give the sequence of three-address code instructions corresponding to each of the arithmetic expressions :

2+3+4+5

(ii) Describe how a for-statement can be systematically turned into a corresponding while statement. Does it make sense to use this to generate code ?

(b) (i) Explain global data flow analysis.

(ii) Explain DAG representation.

(c) Discuss loop optimization technique with suitable examples.

- o O o -

3×2 = 6

Pushpa Ram (CS)

[illegible] 11/5/07