Code No: RT31051 **R13** SET - 1

III B. Tech I Semester Regular Examinations, November - 2015

COMPILER DESIGN

(Computer Science and Engineering)

Time: 3 hours Max. Marks: 70

Note: 1. Question Paper consists of two parts (**Part-A** and **Part-B)**
2. Answering the question in **Part-A** is compulsory
3. Answer any **THREE** Questions from **Part-B**

*****

PART –A

1 a) Define preprocessor. What are the functions of pre-processor? [4M]
b) Discuss about the Syntax Error Handling. [4M]
c) Differentiate between shift-reduce and Operator Precedence Parsers. [4M]
d) What are the benefits of intermediate code generation? [3M]
e) What are the various attributes of a Symbol Table? [3M]
f) Mention the issues to be considered while applying the techniques for code optimization. [4M]

PART -B

2 a) Write a regular expression for identifiers and reserved words. Design the transition diagrams for them. [4M]
b) Explain the three general approaches for the implementation of a Lexical analyzer. [8M]
c) Compare compiler and interpreter with suitable diagrams. [4M]

3 a) Why lexical and syntax analyzer are separated out? [3M]
b) Construct the predictive parser for the following grammar [8M]
S -> (L) | a
L ->L,S | S
c) Give the classification of parsing techniques and briefly explain each. [5M]

4 a) Parse the input string **int id,id;** using shift-reduce parser for the grammar [8M]
S -> TL;
T -> int | float
L -> L,id | id
b) Write the steps for the efficient construction of LALR parsing table. Explain with an example. [8M]

5 a) Translate the assignment x := A[y,z] into three address statement. [8M]
b) Define Type Checker. Write down the specification of a simple Type Checker. [8M]

6 a) How symbol table can be managed? Explain. [8M]
b) Discuss storage allocation for block structured languages. [8M]

7 a) Explain in detail about inter procedural optimization with an example. [8M]
b) Discuss in detail the role of dead code elimination and strength reduction during code optimization of a compiler. [8M]

-000-

Code No: RT31051 **R13** SET - 2

III B. Tech I Semester Regular Examinations, November - 2015

COMPILER DESIGN

(Computer Science and Engineering)

Time: 3 hours Max. Marks: 70

Note: 1. Question Paper consists of two parts (**Part-A** and **Part-B)**
2. Answering the question in **Part-A** is compulsory
3. Answer any **THREE** Questions from **Part-B**

*****

PART –A

1 a) Briefly describe about the Lexical errors. [3M]
b) What are the functions used to create the nodes of syntax trees? [4M]
c) What are the three techniques for constructing LR parsing table? [4M]
d) Discuss the evaluation of semantic rules. [4M]
e) List the characteristics of peephole optimization. [4M]
f) Give the criteria for achieving machine independent code optimization. [3M]

PART -B

2 a) Write regular expressions for the set of words having a,e,i,o,u appearing in that order, although not necessarily consecutively. [4M]
b) Construct NFA equivalent to regular expression $r= (a + b)^*ab$. [8M]
c) Give general format for LEX program. [4M]

3 a) Show that the grammar S -> 0S1| SS |ϵ is ambiguous. [3M]
b) Explain the Non-Recursive predictive parsing with an example. [8M]
c) What are the limitations of recursive descent parser? [5M]

4 a) Write the steps for the construction of CLR parsing table. [8M]
b) Explain the compaction of LR parsing tables with an example. [8M]

5 a) Write the quadruple, triple, indirect triple for the expression [8M]
-(a*b) + (c+d)-(a+b+c+d)
b) Write an algorithm for constructing the dependency graph for a given parse tree. [8M]

6 a) Construct basic blocks, data flow graph and identify loop invariant statements for the following: [8M]

```
for (i=1 to n)
{
   j=1;
   while (j<=n)
   {
      A=B*C/D;
       j=j+1;
    }
}
```

b) Explain how an activation record is related with runtime storage organization. [8M]

7 a) Explain in detail about the instruction scheduling with an example. [8M]

b) What are the principle sources of optimization? Give the classification of code optimization. [8M]

-000-

Code No: RT31051 **R13** SET - 3

III B. Tech I Semester Regular Examinations, November - 2015

COMPILER DESIGN

(Computer Science and Engineering)

Time: 3 hours Max. Marks: 70

Note: 1. Question Paper consists of two parts (**Part-A** and **Part-B)**
2. Answering the question in **Part-A** is compulsory
3. Answer any **THREE** Questions from **Part-B**

*****

PART –A

1 a) Write a LEX program to identify comments in the program. [4M]

b) Consider the CFG S -> SS+|SS*|a . Derive the string **aa+a*** from the given CFG and construct a parse tree for this string. [5M]

c) Differentiate between LR and LL Parsers. [3M]

d) What are the different types of three address statements? [3M]

e) Compare deep access and shallow access. [3M]

f) List the properties of optimizing compilers. [4M]

PART -B

2 a) State the reasons for separating Lexical analysis and Syntax analysis [4M]

b) Describe the lexical errors and various error recovery strategies with suitable examples. [8M]

c) Write a regular expression for relation operators. Design a transition diagram for them. [4M]

3 a) What is left recursion and left factoring? [3M]

b) Verify whether the following grammar is LL(1) or not? [8M]

$E \rightarrow E + T \mid T$
$T \rightarrow T * F / F$
$F \rightarrow (F) \mid a \mid b.$

c) Discuss about error recovery strategies in predictive parsing. [5M]

4 a) Construct the collection of LR(0) item sets and draw the goto graph for the grammar S -> S S | a | ϵ. Indicate the conflicts (if any) in the various states of the SLR parser. [8M]

b) Explain the process of handling “Dangling-ELSE” ambiguity. [8M]

5 a) Construct the syntax tree and postfix notation for the expression (a+ (b*c)) ↑d-e / (f+g). [8M]

b) Explain in detail how an L-attributed grammar can be converted into a translation scheme. [8M]

6 a) Discuss in detail about the Reference counting garbage collectors. [8M]

b) Explain reducible and non-reducible flow graphs with an example. [8M]

7 a) Explain the role of semantic preserving transformations and dominators in code optimization. [8M]

b) Explain with suitable example various sources of loop optimization. [8M]

-000-

Code No: RT31051 **R13** SET - 4

III B. Tech I Semester Regular Examinations, November - 2015

COMPILER DESIGN

(Computer Science and Engineering)

Time: 3 hours Max. Marks: 70

Note: 1. Question Paper consists of two parts (**Part-A** and **Part-B)**
2. Answering the question in **Part-A** is compulsory
3. Answer any **THREE** Questions from **Part-B**

*****

PART –A

1 a) Write the regular expression for the language accepting the strings which are starting with 1 and ending with 0, over the set ∑ = {0,1}. [4M]
b) What are the goals of error handler in a parser? [4M]
c) List the properties of LR parser. [4M]
d) Write the need the Semantic analysis. [3M]
e) Describe the structure of entries in symbol table. [4M]
f) Compare local optimization with global optimization. [3M]

PART -B

2 a) Draw a block diagram of phases of a compiler and indicate the main functions of each phase. [8M]
b) Define lexeme, token and pattern. Identify the lexemes that make up the tokens in the following program segment. Indicate corresponding token and pattern. [8M]

```
void swap(int i, int j)
{
int t;
t=i;
i=j;
j=t;
}
```

3 a) What is an LL(1) grammar? When the grammar is said to be LL(1) grammar? [3M]
b) Design a non-recursive predictive parser for the following grammar. [8M]
S -> AaAb | BbBb
A -> e
B -> e
c) Discuss how Brute-Force approach operates in top down parsing. [5M]

4 a) Draw the structure of LR parser. [3M]
b) Compute closure(I) and goto(I) for the grammar [8M]
S -> Aa| bAc| Bc| bBa
A -> d
B -> d
c) Compare bottom up approaches of parsing with all top down approaches. [5M]

5 a) Construct the syntax tree and draw the DAG for the expression (a*b) + (c-d) * (a*b) + b. [8M]

b) Write Syntax directed definition for constructing syntax tree of an expression derived from the grammar [8M]
E -> E + T | E – T | T
T -> (E) | id | num

6 a) What is Peephole optimization? Explain its characteristics. [8M]

b) Explain with an example optimization of Basic blocks. [8M]

7 a) Discuss how copy propagation can be done using data flow equation. [8M]

b) Explain in detail the procedure that eliminates global common sub-expression. [8M]

-000-