Registration No :

Total Number of Pages : 02 **B.Tech**
PCS6I102

6th Semester Regular / Back Examination 2018-19
COMPILER DESIGN
BRANCH : CSE
Max Marks : 100
Time : 3 Hours
Q.CODE : F201

Answer Question No.1 (Part-1) which is compulsory, any eight from Part-II and any two from Part-III.

The figures in the right hand margin indicate marks.

Part- I

Q1 **Only Short Answer Type Questions (Answer All-10)** **(2 x 10)**

a) Is macro processing a phase in compilation? Justify your answer

b) List the various error recovery strategies for a lexical analysis.

c) Define left recursion. Eliminate left recursion from the following grammar
E → E+T | T
T → T*F | F
F → (E) | id

d) Explain the purpose of semantic analysis in a compiler.

e) List the rules for computing Follow set of a grammar

f) What optimization can you propose for the following code
a := b*c;
x := b*c +5;

g) Mention the conflicts that occur in shift-reduce parser.

h) Mention the strategies of storage allocation.

i) Draw the annotated parse tree for “int a, b, c;”
D → T L; | L.inh = T.type
T →int | T.type = integer
T → float | T.type = float
L → L1, id | L1.inh = L.inh
| addType (id.entry, L.inh)
L → id | addType(id.entry, L.inh)

j) Why symbol table is required? List various attributes of symbol table.

Part- II

Q2 **Only Focused-Short Answer Type Questions- (Answer Any Eight out of Twelve)** **(6 x 8)**

a) Construct the NFA that consists of all strings of a’s and b’s where third symbol from th right end is ‘a’. convert the NFA to corresponding DFA.

b) Define Context free grammar. Find out the context fee grammar for the following languages that consists of all the strings of a’s and b’s where
i) Every string starts and ends with the same symbol.
ii) $L=\{a^m b^n c^p \mid n= m+p \text{ and } m, n, p \geq 0\}$

c) State the various phases of a compiler, indicating the inputs and outputs of each phase in translating the statement “position = initial + rate * 60”

d) Explain various issues associated with grammars in top-down parsing with suitable example.

e) Explain different type expressions with example.

f) Compare the different implementations of three address codes with examples

g) What is back patching. Generate three address code for the following Boolean expression using back patching
a < b or c > d and e < f

h) Mention the job of code generator. Explain the simple code generation using stack allocation.

i) Explain peephole optimization.

j) Write an Syntax directed translation to convert a binary number to decimal number. For example, when 101.101 is given as an input, it outputs 5.625. Illustrate the Syntax Directed Translation while parsing the input given in example.

k) Distinguish between S-attributed, I-attributed and L-attributed definition with suitable example.

l) Explain how scope rules and the block structure of aprogramming language influence symbol table organizationstrategies.

Part-III

Only Long Answer Type Questions (Answer Any Two out of Four)

Q3 Consider the following grammar **(16)**
E → E+T | T
T → T*F | F
F → (E) | id

a) Find the CLR parser for the above grammar.
b) Show the parsing of the string “((id + id) * id) + id” using the parsing table constructed above.

Q4 What are the various intermediate forms? Mention its types. How would you implement the three address statements? Generate intermediate code for the following program fragment. Assume there are four bytes per word **(16)**

```
        sum=0;
        for(i=1;i<=20;i++)
sum = sum + a[i] + b[i];
```

Q5 Consider the following program segment: **(16)**

```
Prod = 0;
I=1;
do
{
Prod = Prod + A[I] * B[I];
I = I + 1;
} while (I ≤ 20)
```

Assume that A and B are allocated static storage and there are 4 bytes per word in byte addressable manner. Perform the following tasks on the above program fragment.

a) Generate three address code.
b) Partition into basic blocks
c) Construct flow graphs on basic blocks
d) Perform loop optimization using code motion, loop invariant elimination and induction variable elimination.

Q6 What is an activation record? Draw diagram of General Activation record and explain the purpose of different fields of an activation record. **(16)**