DHARMSINH DESAI UNIVERSITY, NADIAD
FACULTY OF TECHNOLOGY
B.TECH. SEMESTER VII [CE]
SUBJECT: (CE-718) COMPILER CONSTRUCTION

Examination	**: Second Internal**	**Seat No**	**: CE-84**
Date	**: 06/09/2019**	**Day**	**:**
Time	**: 01:45 P.M.-03:00 P.M.**	**Max. Marks**	**: 36**

INSTRUCTIONS:
1. Figures to the right indicate maximum marks for that question.
2. The symbols used carry their usual meanings.
3. Assume suitable data, if required & mention them clearly.
4. Draw neat sketches wherever necessary.

Q.1 Do as directed.

(a) For a CFG: $\hat{S} \rightarrow S\,S+ \mid S\,S* \mid a$ identify if is in LL(1) or not ? **[2]**

(b) Remove Left recursion from the given grammar (if present) **[2]**

$E \rightarrow E+T \mid T \;;\; T \rightarrow T*F \mid F \;;\; F \rightarrow (E) \mid id$

(c) Remove left factoring from the given grammar (if present) **[2]**

$S \rightarrow i\,E\,t\,S \mid i\,E\,t\,S\,e\,S \mid a$

$E \rightarrow b$

(d) Consider the following code. **[2]**

```
int fact(int n)
{
  return (n*fact(n-1));
}
```

If I call fact(4) then show the static link and dynamic link on the Run time stack.

(e) Justify with example: Static scope based language paradigm is more intuitive than Dynamic scope based paradigm. **[2]**

(f) Compute the total storage space required for the following nested block structure with the size of data member as given in the comments. **[2]**

```
B1{a,b,c: /*sizes 10 20 20;*/
   B2{d,e,f: /*sizes 100 180 40}
   B3{g,h,i;  /*sizes 50 20 10*/
        B4{j,k,l; /*sizes 70 150 20*/}
        B5{m,n,p;/*sizes 20 50 30*/}
    }
   }
```

Q.2 Answer any *TWO* of the following.

(a) Give a detailed trace of run time stack during the execution of a block structured segment of program as given below. **[6]**

```
1  BBLOCK:
       REAL X,Y,STRING NAME;
       :
2      M1.PBLOCK(INTEGER IND);
           INTEGER X;
           :
           CALL M2(IND+1);
           :
       END M1;
3      M2.PBLOCK(INTEGER J);
           :
4          BBLOCK
               ARRAY INT F(J);LOGICAL TEST1;
               :
           END;
       END M2;
       :
       CALL M1(X/Y);
   END;
```

(b) (i) What is Boundary Tag Method? Why is it used? Explain each of the 4 case with necessary figure. **[4]**

(ii) Assume that reference x points to object O and reference y points to object P. Assume current reference count values for O and P are 1 and 2 respectively. [2]

- What will be the value of reference count values for assignment of type x=y?
- When will space allocated to the object be freed? After x=y is there any object which is possible candidate of garbage collection? Justify.

(c) Explain Mark and Sweep based Garbage Collector with algorithm and example. [6]

Q.3 Answer the following.

(a) Write an algorithm for recursive descent parser for the grammar: [8]

f → (e) | id ; t → f * t | f ; e → t + e | t

(b) Consider CFG as : [4]

Stmt → if Expr then Stmt | if Expr then Stmt else Stmt | S1 | S2.

Expr → E1 | E2

according to this grammar generate all possible parse trees for the statement "if E1 then if E2 then S1 else S2".

OR

(a) C → P F class id X Y [8]

P → public | ϵ

F → final | ϵ

X → extends id | ϵ

Y → implements I | ϵ

I → id J

J → , I | ϵ

Find FIRST and FOLLOW sets of all non terminals.

(b) Generate predictive parsing table for the grammar given in above question. [4]