



DHARMSINH DESAI UNIVERSITY, NADIAD
FACULTY OF TECHNOLOGY
B.TECH. SEMESTER VI [IT]
SUBJECT: (IT608) LANGAUGE TRANSLATOR

Examination :Third Sessional **Seat No. : _____**
Date : 24/3/2014 **Day : Monday**
Time : 12.45 to 2.00 **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) Match the following:

[2]

List I

List II

- | | |
|--------------------------|------------------------------------|
| (a) Activation record | (1) Life time of binding |
| (b) Reference counts | (2) Subroutine call |
| (c) Three address code | (3) Garbage collection |
| (d) Scope of declaration | (4) Code-improving transformations |

(b) For following program, which phases of “C” compiler would generate errors? [2]

```
main () { int i =0;
        do { if(i>10)
              continue;
              i++;
        } while(i<20)
        Printf(“\n i = %d”,i);
    }
```

(c) Justify:- “With reference to 3-address code representation methods, Utility of indirect triples is same as that of quadruples” [2]

(d) Show the Syntax tree and DAG representation for the expression $w - w * (x + y) - (x + y) * z$. [2]

(e) If you were to design a compiler, would you provide optimization phase? Why? [1]

(f) In _____ parameter passing method, the r-value of actual parameter is passed on procedure call, and the l-value of the formal parameter is copied to actual parameter on return from procedure. [copy-restore/call by reference] [1]

(f) Recursion is not possible in which of the following storage allocation. [1]
(A) Static (B) Stack (C) Heap (D) None of these.

(g) In _____ type of Intermediate code representation, common sub expression is been shown only once in a tree while in _____ type of Intermediate code representation, explicit names are given to each computation result. [syntax tree/ DAG/ 3- address code/postfix form] [1]

Q.2 Attempt *Any Two* from the following questions.

[12]

(a) The following program fragment is written in a programming language that allows global variables and does not allow nested declarations of functions. [6]

```
global int j=10;
void f1(x) {printf(“in f1 = %d”,j);}
void f2(x) {int j=20; f1(x);}
int main() { f1(j+1); f2(j+1); f1(j+1);}
```

(I) If the programming language uses static scoping and pass by value parameter Passing mechanism, what are the values printed by the above program?

(II) If the programming language uses dynamic scoping and pass-by-reference parameter passing mechanism, what are the values printed by the above program?

(b) Consider following program, in block structured pseudo –language with lexical scoping and nesting of procedures permitted. [6]

```
Proc M() {
    Proc p()
    {
        int x,y,z;
        proc q()
        {
            proc r() { z=p(); }
            y=r();
        }
        x=q();
    }
    P();
}
```

Consider the calling chain :

main called M, M called p, p called q, q called r, r called p

Show contents of runtime memory, with clear details if access links used.

(c) Generate target code for following C statements .Assume 2 registers are available in target machine. $x = (a - b) / (c + ((a - b) / c))$ [6]

Q.3 (a) Find and explain in brief optimizations possible for following program fragment. [6]

```
main()
{
    int a=10,b=20,c,d,e=30,f,x=0;
    for(i=0;i<10;i++)
    {
        c=a+b; d=10*20; f=50*e; sum=sum+i;
        a=x+y; b=a; d=b;
    }
    if(a<b)      {c=x+y; b=c; }
    if(x == 1)   { c=a+b ; x++;}
```

Note: clearly state the names of optimizations performed.

(b) Write short note on: error response and error recovery strategies. [6]

OR

Q.3 (a) Give 3-Address IC for following pseudo code. [6]

```
Begin
  While (a>b and b>c) do
    Begin
      x=y+z
      a=a-b
    End
    x=y-z
  End
```

And state atleast two advantages of intermediate code.

(b) Write short note on: Storage allocation strategies for run-time environment. [6]