Total No. of Questions :6]

SEAT No. :

P108

[Total No. of Pages :3

APR. -16/TE/Insem. - 45
T.E. (Information Technology)
SYSTEMS PROGRAMMING
(2012 Pattern) (314450) (Semester - II)

Time : 1Hour] *[Max. Marks :30*

Instructions to the candidates:

1) *Answer Q1 or Q2, Q3 or Q4, Q5 or Q6.*
2) *Neat diagrams must be drawn wherever necessary.*
3) *Figures to the right indicate full marks.*
4) *Assume Suitable data if necessary.*

Q1) a) Generate symbol table, literal table, pool table, Intermediate code and Target Code for the given assembler program. Assume a hypothetical instruction set with every instruction of length 1. **[8]**

```
      START   100
A     DS      05
      LOAD    A
      ADD     AREG, ='5'
      MULT    BREG, ='10'
      TRANS   L
L2    PRINT   L1
      LTORG
L     ADD     AREG, ='5'
      SUB     BREG, ='15'
      ADD     B
B     EQU     L + 10
      ORIGIN  L2+20
L1    DS      5
```

P.T.O.

```
C    DC 10
     STOP
     END
```

b) Define the term forward reference and explain w.r.t. the example in Q1.a) **[2]**

OR

Q2) a) Enlist and explain various data structures required for design of 2-pass assembler. **[4]**

b) Perform single pass macroprocessor algorithm and generate expanded code for the given code: **[6]**

```
MACRO
M2
AR 2,3
SR 4,4
MACRO
M1  &S1
AR  3,3
&S1 X
SR  4,4
L   1, = F'4'
DC  A(X)
L   2, = V(&SI)
BALR 14,15
MEND
MEND
START
AR  1,1
M2
SR  2,2
M1  PQR
END
```

Q3) a) Define loader and enlist the basic functions of loader. [4]

b) Explain transfer vector and relocation bits w.r.t BSS loader and compare BSS loader with absolute loader. [6]

OR

Q4) a) What information must be supplied by an assembler to the direct linking loader (DLL). Explain the significance of this information w.r.t design of DLL. [5]

b) Explain the design of absolute loader. [5]

Q5) a) Explain the first three phases of compiler w.r.t the following statement: [5]

r = a – 10 / (c * d ^ e)

Note: ^ is exponentiation operator.

b) For the given piece of code perform lexical analysis and generate various tables, void main ( ) [5]

```
{
   float rad, area;
   int i = 0;
   clrscr ( );
   printf("Welcome to C programming!");
   i++;
   printf("%d", i);
   getch ( ):
}
```

OR

Q6) a) Using the RE to DFA algorithm generate the DFA for the given regular expression [8]

(a/b) * a * b

b) Lexical analyzer builds an UST. What is the need of this table? [2]