(3 Hours) **Total Marks: 80**

N.B: (1) Question No. 1 is compulsory.
(2) Attempt any three questions out of the remaining five questions.
(3) Make suitable assumptions wherever necessary.

==========

Q.1. A. What is three-address code? Generate three-address code for – **5**

```
while (a<b) do
    if(c<d) then
        x:=y+z
    else
        x:=y-z
```

B. Compare between Compiler and Interpreter. **5**

C. Explain absolute loader. State its advantages and disadvantages. **5**

D. Discuss with example 'forward reference'. **5**

Q.2. A. Construct SLR parser for the following grammar and parse the input "( )( )": **S → (S)S | ε.** **10**

B. State and explain with examples, different types of statements used in assemblers with respect to system programming. **10**

Q.3. A. Explain the concept of basic blocks and flow graph with example the three-address code. **10**

B. Explain with help of a flowchart, the first pass of two-pass macro processor. **10**

Q.4. A. Explain the phases of a compiler. Discuss the action taken in various phases to compile the statement:
a=b*c+10, where, a, b, c are of type real. **10**

B. Write short note on: **10**
(i) Syntax-directed Translation, (ii) Macro facilities

Q.5. A. What is code optimization? Explain with example, the following code optimization techniques: **10**
(i) Common sub-expression elimination (ii) Code motion
(iii) Dead code elimination (iv) Constant propagation

B. Explain Direct Linking Loader in suitable example. **10**

Q.6. A. Test whether following grammar is LL(1) or not. If it is LL(1), construct parsing table for the same: **10**

$S \rightarrow 1AB|\varepsilon$
$A \rightarrow 1AC|0C$
$B \rightarrow 0S$
$C \rightarrow 1$

B. Draw and explain the flowchart of Pass-I of two pass assembler with suitable example. **10**
