POKHARA ENGINEERING COLLEGE

Final Assessment

Level: Bachelor | Semester – Spring | Year : 2025
Programme: B.E Computer | Full Marks: 100
Course: Compiler Design | Pass Marks: 45
Time : 3hrs.

Candidates are required to give their answers in their own words as far as practicable.

The figures in the margin indicate full marks.

Attempt all the questions.

1. a) Explain the phases of a compiler with a neat block diagram. Describe the functionality of each phase in brief. 3+5
   b) Differentiate between a compiler and an interpreter. What are the four main types of compiler? 2+5
2. a) What is the Mathematical Model of Finite Automata? Draw a non deterministic finite automate which accept 00 and 11 at the end of a string containing 0, 1 in it, e.g., 01010100 but not 000111010 3+5
   b) Explain about Thomson's Construction. 7
3. a) What are the steps for transforming a grammar for top down parsing? Explain with examples. 7
   b) Construct LL(1) parsing table for following grammar. 8

```
E --> TE'
E' --> +TE' | ε
T --> FT'
T' --> *FT' | ε
F --> id | (E)
```

*ε denotes epsilon

4. a) Describe different intermediate code forms used in compiler design. Illustrate with an example the generation of three-address code. 7
   b) What is the role of symbol tables in various phases of compiler design? Generate symbol table for the following line of code: 8

```
Int x;
Float y, sum;
Sum = x+y;
```

5. a) What are the Issues in the design of a code generator? 7
   b) Explain about register allocation in code generation? Perform register assignment for the following: 8

```
1: a = 10
2: b = 20
3: c = a + b
4: d = c + 1
5: e = d + b
```

6. a) What is code optimization? Explain common optimization techniques like constant folding, dead code elimination, and strength reduction. 8
   b) For the following grammar construct LR(1) parsing table: 7

$S \rightarrow AaAb$
$S \rightarrow BbBa$
$A \rightarrow \varepsilon$
$B \rightarrow \varepsilon$

7. Write short notes on **any two**: 2×

a) Peephole optimization

b) Dynamic Programming.

c) Loop optimization.