YMCA UNIVERSITY OF SCIENCE& TECHNOLOGY, FARIDABAD

B.Tech., V SEMESTER

System Software Design (CE-311)

Time: 3 Hours | Max. Marks:60

Instructions:
1. *It is compulsory to answer all the questions (2 marks each) of Part -A in short.*
2. *Answer any four questions from Part -B in detail.*
3. *Different sub-parts of a question are to be attempted adjacent to each other.*

PART -A

Q1 (a) What is the need of system programming? (2)

(b) How the macros are used in system Programming? (2)

(c) What is a regular expression? (2)

(d) Distinguish between NFA and DFA with an example. (2)

(e) Why the binders are used? Explain in brief. (2)

(f) Evaluate the post fix expression ab+c* with a,b and c have values1,3 and 5. (2)

(g) What are the compiler writing tools? (2)

(h) Write the contents of a symbol table. (2)

(i) What is a DAG representation? (2)

(j) How the register allocation method is used? (2)

PART -B

Q2 (a) How the assembler, linker and loader are in relation? Explain. (5)

(b) What are the "compile and go loader" and absolute loader schemes? (5)

Q3 (a) Explain the method for construction of NFA from a regular expression. (5)

(b) What are the debug monitor and text editor software tools? Explain (5)

Q4 (a) What are the different phases of a compiler? Explain in details. (5)

(b) Consider the following grammar (5)

$E \rightarrow E+T/T$

$T \rightarrow T*F/F$

$F \rightarrow (E)/id$

Write the terminal and non-terminals and eliminate the immediate left recursion.

Q5 (a) What is intermediate code generation? Explain three address codes. (5)

(b) Consider the grammar (5)

$S \rightarrow iCtSS/a$

$S' \rightarrow eS/$

$C \rightarrow b$ Construct the predictive parsing table

Q6 (a) Consider the context free grammar (5)

$S \rightarrow S+S|SS|(S)|S^*|a$

and the string is (a+a)*a

(i) Give the leftmost derivation for the string.

(ii) Give the rightmost derivation for the string.

(iii) Give a parse tree for the string

(b) What are the LR parsers? Explain in brief. (5)

Q7 Explain the principles sources of optimization and peephole optimization (10)

****