B.E. (Full Time) - END SEMESTER EXAMINATIONS – NOV/DEC 2021

Computer Science and Engineering

V SEMESTER

CS6109 Compiler Design

Time: 3 Hrs | Regulation RUSA 2018 | Max Marks: 100

PART - A (10 x 2 = 20 Marks)

(Answer ALL Questions)

1. Which phases of compiler forms the front end of the compiler and what is the outcome of the front end?
2. What are the roles of compiler construction tools?
3. Differentiate lexemes from tokens with example.
4. How does a lexical analyzer detect errors?
5. What is the outcome of the translation of the declarations in any program by a compiler?
6. What is the significance of look ahead in the LR(1) items?
7. Define inherited translation, synthesized translation and give example for both types of translations.
8. Write the translation scheme for constructing syntax tree from arithmetic expression.
9. Which variables should be kept live and what is its advantage?
10. What is meant by a peephole and what optimizations are applicable on a peephole?

PART - B (8 x 8 = 64 marks)

(Answer any EIGHT questions only)

11. Explain the five phases of any compiler, accepting high level program and producing object program, with specific example.
12. Find the minimal state DFA for the regular expression (a+b)(a+b)*ab.
13. Consider the grammar with production rules B → B + B | B B | ¬B | (B) | true | false. Remove left recursion in this grammar and show this grammar is LL(1) by constructing predictive parsing table.

14. Explain the error recovery strategies used by compilers. Show how the predictive parser recovers from errors while parsing strings?

15. Show that the CFG S → Aa | bAc | Bc | bBa is not LALR(1).

16. Generate intermediate code for the boolean expression “P<Q and R<S or T<U” and give the necessary translations.

17. Translate A [i] := B [i, A [i+j*k]] into three address statements.

18. Translate the following code segment into three address code.

```
while (a < b ) if (a%2 == 0) a = + 1 else a = a + 2
```

19. Explain how symbol tables are organized at runtime.

20. Discuss the principal sources of optimization and any three local optimization techniques.

21. Explain the identification of leaders, creation of basic blocks and the creation of flow graphs from the given list of three address statements.

22. Apply the code generation algorithm to generate code for the following assignment statement. W := (A + B) + (B + A) + (C + C).

PART - C (2 x 8 = 16 marks)

(Answer ALL Questions)

23. Tabulate the lexeme, pattern and token from the following program segment.

```
token = (gender == ‘F’) ? (regNo && 0xff) : (regNo >> 4) && 0xff;
while (true) {    rem = n % i;
                  if (!rem) return 0;
                  if (i>n/2) return 1;
                  i += 1;
              }
```

24. Construct the left most derivation and right most derivation for the string “i n t i n t i n t r = n + n + n e r = n” using the following CFG.

S → i E t S | i E t S e S | A
A → r = E
E → E + T | T
T → n