Daffodil International University

Department of Computer Science and Engineering

Faculty of Science & Information Technology

Final Examination, Summer 2021 @ DIU Blended Learning Center

Course Code: CSE 331 (Day), Course Title: Compiler Design

Level: 3 Term: 3 Section: All

Instructor: SMTS Modality: Open Book Exam

Date: Tuesday 24 August, 2021 Time: 09:00 AM - 12:30 PM

Three hours and Thirty Minutes (3:30 Hours) Marks: 40

The figures in the right margin indicate the full marks.
All portions of each question must be answered sequentially.

Q1. (a) The following three address code is to implement the **Quicksort Algorithm**. Considering the following code answer the following questions.

1. i = m-1	**11. t5 = a[t4]**	**21. a[t10] = x**
2. j = n	**12. if t5>v goto (9)**	**22. goto (5)**
3. t1 = 4 * n	**13. if i>=j goto (23)**	**23. t11 = 4 * i**
4. v= a[t1]	**14. t6 = 4 * i**	**24. x = a[t11]**
5. i = i+1	**15. x = a[t6]**	**25. t12 = 4 * i goto (19)**
6. t2 = 4 * i goto (2)	**16. t7 = 4 * i**	**26. x = a[t11]**
7. t3 = a [t2]	**17. t8 = 4 * j**	**27. t12 = 4 * I goto (15)**
8. If t3<v goto (5)	**18. t9 = a[t8]**	**28. a [t12] = t14**
9. j = j -1	**19. a[t7] = t9**	**29. t15 = 4 * m**
10. t4 = 4 * j	**20. t10 = 4 * j**	**30. a[t15] = x**

[6]

I. Which lines of the above code are leaders by leader selection rule 2?
II. Which lines of the above code are leaders by leader selection rule 3?
III. How many basic blocks are present in the above code?
IV. Any instructions which are selected as leader more than 2 times?

(b) Draw the Basic block & follow graph for above mentioned instructions in ***question no (1a)***. **[4]**

Q2. (a) Draw the **Syntax Tree** & **Directed Acyclic Graph (DAG)** for the following expressions. **[4]**

I. X = a * c + (a * (c – a + d))
II. ((t + m) – ((t + m) * (t - m))) + ((p + q) * (p -q))

(b) For the expressions stated in **Q2(a)** produce **Three Address Code, Quadruples** and **Indirect Triples** data structure. Consider “62” as the starting statement number for indirect triples. **[6]**

Q3. **(a)** Consider the following CFG: **[6]**

E → MM
M → FN | cF | aM | cM | FM
F → ab | ad | N
N → ra | ar | ε

What will be the output of the **FIRST()** and **FOLLOW()** functions considering the non-terminals as parameter? Construct the **LL(1)** predictive parsing table using the outputs of **FIRST()** and **FOLLOW()** functions.

(b) Apply left factoring techniques to eliminate non-determinism from the following CFG. **[4]**

P → rPPssP | rPPsPr | rPP | rP | sPs | r
E → id | ε

Q4. **(a)** Consider the following CFG and answer the following questions. **[6]**

E → E + T | T
T → T * F | F
F → (E) | id

Determine the augmented form of the above CFG also Produce **LR(0)** diagram and **Canonical Table** from the above grammar.

(b) Consider the following CFG and answer the following questions. **[4]**

rexpr → rexpr + rterm | rterm
rterm → rterm * rfactor | rfactor
rfactor → rfactor * rprimary | rprimary
rprimary → a | b | c

Construct the right recursive grammar from the above CFG. Consider "**a * b + c**" as input string of the above CFG. Draw two different parse tree using the CFG before and after eliminating left recursion.