



Daffodil International University

Faculty of Science & Information Technology

Department of Computer Science & Engineering

Final Examination, Summer 2025

Course Code: CSE313, Course Title: Compiler Design

Level: 3 Term: 1 Batch: 64

Time: 02:00 Hrs.

Marks: 40

Answer ALL Questions

[The figures in the right margin indicate the full marks and corresponding course outcomes. All portions of each question must be answered sequentially.]

1.	a)	Consider the following grammar to produce LR(0) parser and Canonical Table. $S \rightarrow N@D.T$ $N \rightarrow a b$ $D \rightarrow Gm m$ $G \rightarrow e y g$ $T \rightarrow c d$	[8]	CO2
	b)	<pre>int main() { int x = 5, y = 10; int p = x + y; int q = x + y; int z = p * 2; int r = z; int w = r + 1; r = 20; return w; }</pre> Explain the techniques of Code Optimization(s) that can be applied to the code and rewrite the code after applying these optimization(s) to improve its efficiency.	[2]	
2.	a)	Develop FIRST and FOLLOW functions for the following production rules and determine the LL(1) Parsing Table: $S \rightarrow S+M M$ $M \rightarrow M*R R$ $R \rightarrow aSb int$	[8]	CO2
	b)	State the rules of the FIRST() and FOLLOW() functions.	[2]	
3.	a)	Consider the following arithmetic expression: $(a + b) * (c - d) + e * f - (g + h) * i$ Apply intermediate code generation techniques to evaluate the expression. - Apply your knowledge to write the Three Address Code (TAC) for the expression. - Generate the corresponding Quadruples representation. - Analyze the structure and produce the Indirect Triples representation where you have 27, 29, 31, 33, 67, and 91 available memory in STACK.	[6]	CO3

	b) Consider the following arithmetic expression: $a + b * (c - d) - e + b * (c - d)$ i. Construct the Syntax Tree ii. Construct the DAG (Directed Acyclic Graph)	[4]																																																													
4.	<table border="1"> <tbody> <tr><td>1</td><td>i = m-1</td><td>16</td><td>t₇ = 4 * i</td></tr> <tr><td>2</td><td>j = n</td><td>17</td><td>t₈ = 4 * j</td></tr> <tr><td>3</td><td>t₁ = 4 * n</td><td>18</td><td>t₉ = a[t₈]</td></tr> <tr><td>4</td><td>v = a[t₁]</td><td>19</td><td>a[t₇] = t₉</td></tr> <tr><td>5</td><td>i = i+1</td><td>20</td><td>t₁₀ = 4 * j</td></tr> <tr><td>6</td><td>t₂ = 4 * i</td><td>21</td><td>a[t₁₀] = x</td></tr> <tr><td>7</td><td>t₃ = a[t₂]</td><td>22</td><td>goto (5)</td></tr> <tr><td>8</td><td>if t₃ < v goto (5)</td><td>23</td><td>t₁₁ = 4 * i</td></tr> <tr><td>9</td><td>j = j-1</td><td>24</td><td>x = a[t₁₁]</td></tr> <tr><td>10</td><td>t₄ = 4 * j</td><td>25</td><td>t₁₂ = 4 * i</td></tr> <tr><td>11</td><td>t₅ = a[t₄]</td><td>26</td><td>t₁₃ = 4 * n</td></tr> <tr><td>12</td><td>if t₅ > v goto (9)</td><td>27</td><td>t₁₄ = a[t₁₃]</td></tr> <tr><td>13</td><td>if i >= j goto (23)</td><td>28</td><td>a[t₁₂] = t₁₄</td></tr> <tr><td>14</td><td>t₆ = 4 * i</td><td>29</td><td>t₁₅ = 4 * n</td></tr> <tr><td>15</td><td>x = a[t₆]</td><td>30</td><td>a[t₁₅] = x</td></tr> </tbody> </table> Answer the following questions from the above three address code-	1	i = m-1	16	t ₇ = 4 * i	2	j = n	17	t ₈ = 4 * j	3	t ₁ = 4 * n	18	t ₉ = a[t ₈]	4	v = a[t ₁]	19	a[t ₇] = t ₉	5	i = i+1	20	t ₁₀ = 4 * j	6	t ₂ = 4 * i	21	a[t ₁₀] = x	7	t ₃ = a[t ₂]	22	goto (5)	8	if t ₃ < v goto (5)	23	t ₁₁ = 4 * i	9	j = j-1	24	x = a[t ₁₁]	10	t ₄ = 4 * j	25	t ₁₂ = 4 * i	11	t ₅ = a[t ₄]	26	t ₁₃ = 4 * n	12	if t ₅ > v goto (9)	27	t ₁₄ = a[t ₁₃]	13	if i >= j goto (23)	28	a[t ₁₂] = t ₁₄	14	t ₆ = 4 * i	29	t ₁₅ = 4 * n	15	x = a[t ₆]	30	a[t ₁₅] = x		CO3
1	i = m-1	16	t ₇ = 4 * i																																																												
2	j = n	17	t ₈ = 4 * j																																																												
3	t ₁ = 4 * n	18	t ₉ = a[t ₈]																																																												
4	v = a[t ₁]	19	a[t ₇] = t ₉																																																												
5	i = i+1	20	t ₁₀ = 4 * j																																																												
6	t ₂ = 4 * i	21	a[t ₁₀] = x																																																												
7	t ₃ = a[t ₂]	22	goto (5)																																																												
8	if t ₃ < v goto (5)	23	t ₁₁ = 4 * i																																																												
9	j = j-1	24	x = a[t ₁₁]																																																												
10	t ₄ = 4 * j	25	t ₁₂ = 4 * i																																																												
11	t ₅ = a[t ₄]	26	t ₁₃ = 4 * n																																																												
12	if t ₅ > v goto (9)	27	t ₁₄ = a[t ₁₃]																																																												
13	if i >= j goto (23)	28	a[t ₁₂] = t ₁₄																																																												
14	t ₆ = 4 * i	29	t ₁₅ = 4 * n																																																												
15	x = a[t ₆]	30	a[t ₁₅] = x																																																												
	a) i. Which lines in the code qualify as leaders by leader selection rule 2? ii. Is / Are there any instruction(s) designated as leaders more than twice? iii. What is the total number of basic blocks identified in the code?	[6]																																																													
	b) Draw the flow graph for the above mentioned instructions.	[4]																																																													