



Daffodil International University
Faculty of Science & Information Technology
Department of Computer Science & Engineering
Final Examination, Summer 2025
Course Code: CSE331, Course Title: Compiler Design
Level: 3 Term: 3 Batch: 62

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. $E \rightarrow E+E \mid E * E \mid (E) \mid id$	[8]	CO2
	b)	<pre>int main() { int a = 10, b=2; int c = a + b; int f = a + b; int i = f; while (i < 20) { int temp2 = a * b; i++; } if (c < 0) { i = e + f; } return temp2+i ; }</pre> Relate the optimization techniques that can be applied to the above code. Rewrite the code after applying optimization techniques.	[2]	
2.	a)	Suppose you are part of a compiler design team assigned to develop a predictive parser for a basic arithmetic expression grammar. The grammar is defined as: $E \rightarrow E+T \mid E-T \mid T \quad T \rightarrow T * F \mid T / F \mid F \quad F \rightarrow (E) \mid id$ Apply analytical skills to: - Eliminate left recursion from the given grammar and write the equivalent grammar suitable for predictive parsing. - Develop the predictive parsing table for the modified grammar - When can a grammar be called an LL(1) grammar?	[2] [6] [2]	CO2
3.	a)	Consider the following expression and answer the questions. $speed + distance * (rate + fee) - (limit * speed) + gain / (extra + fee) + distance * limit$ - Represent the expression using Three Address Code (TAC). - Represent the expression using Quadruples. - Implement a Triples data structure to represent the expression.	[6]	CO3
	b)	Using 3(a) equation answer the following questions.	[4]	

		i. Construct the Syntax Tree for the expression. ii. Construct a Directed Acyclic Graph (DAG) for the expression.																																																														
4.	a)	<table><tr><td>1</td><td>t89 = b[t42]</td><td>16</td><td>if t17 > t52 goto (22)</td></tr><tr><td>2</td><td>goto (22)</td><td>17</td><td>t3 = t40 / t65</td></tr><tr><td>3</td><td>a[t82] = t59</td><td>18</td><td>if t42 < t5 goto (11)</td></tr><tr><td>4</td><td>b[t16] = t89</td><td>19</td><td>goto (46)</td></tr><tr><td>5</td><td>if t29 >= t98 goto (16)</td><td>20</td><td>if t17 >= z goto (22)</td></tr><tr><td>6</td><td>t96 = a[t66]</td><td>21</td><td>t89 = a[t52]</td></tr><tr><td>7</td><td>m = t15</td><td>22</td><td>t38 = c[t13]</td></tr><tr><td>8</td><td>t77 = a[t3]</td><td>23</td><td>v = t67</td></tr><tr><td>9</td><td>c[t53] = t16</td><td>24</td><td>if t44 >= t2 goto (26)</td></tr><tr><td>10</td><td>z = t12</td><td>25</td><td>x = t35</td></tr><tr><td>11</td><td>j = t52</td><td>26</td><td>if t14 < t69 goto (19)</td></tr><tr><td>12</td><td>j = t72</td><td>27</td><td>b[t57] = y</td></tr><tr><td>13</td><td>goto (42)</td><td>28</td><td>x = t63</td></tr><tr><td>14</td><td>if t22 == t34 goto (36)</td><td>29</td><td>t82 = b[t54]</td></tr><tr><td>15</td><td>t90 = b[t53]</td><td>30</td><td>v = t40</td></tr></table> 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. Conclude the total number of basic blocks identified from the code?	1	t89 = b[t42]	16	if t17 > t52 goto (22)	2	goto (22)	17	t3 = t40 / t65	3	a[t82] = t59	18	if t42 < t5 goto (11)	4	b[t16] = t89	19	goto (46)	5	if t29 >= t98 goto (16)	20	if t17 >= z goto (22)	6	t96 = a[t66]	21	t89 = a[t52]	7	m = t15	22	t38 = c[t13]	8	t77 = a[t3]	23	v = t67	9	c[t53] = t16	24	if t44 >= t2 goto (26)	10	z = t12	25	x = t35	11	j = t52	26	if t14 < t69 goto (19)	12	j = t72	27	b[t57] = y	13	goto (42)	28	x = t63	14	if t22 == t34 goto (36)	29	t82 = b[t54]	15	t90 = b[t53]	30	v = t40	[6]	CO3
1	t89 = b[t42]	16	if t17 > t52 goto (22)																																																													
2	goto (22)	17	t3 = t40 / t65																																																													
3	a[t82] = t59	18	if t42 < t5 goto (11)																																																													
4	b[t16] = t89	19	goto (46)																																																													
5	if t29 >= t98 goto (16)	20	if t17 >= z goto (22)																																																													
6	t96 = a[t66]	21	t89 = a[t52]																																																													
7	m = t15	22	t38 = c[t13]																																																													
8	t77 = a[t3]	23	v = t67																																																													
9	c[t53] = t16	24	if t44 >= t2 goto (26)																																																													
10	z = t12	25	x = t35																																																													
11	j = t52	26	if t14 < t69 goto (19)																																																													
12	j = t72	27	b[t57] = y																																																													
13	goto (42)	28	x = t63																																																													
14	if t22 == t34 goto (36)	29	t82 = b[t54]																																																													
15	t90 = b[t53]	30	v = t40																																																													
	b)	Draw the flow graph for the above mentioned instructions.	[4]																																																													