

# Faculty of Engineering / Science

## End Semester Examination May 2025

### CS3CO44 / BC3CO68 Compiler Design

<b>Programme</b>	:	B.Tech. / B.Sc.	<b>Branch/Specialisation</b>	:	CS
<b>Duration</b>	:	3 hours	<b>Maximum Marks</b>	:	60

**Note:** All questions are compulsory. Internal choices, if any, are indicated. Assume suitable data if necessary.  
 Notations and symbols have their usual meaning.

#### Section 1 (Answer all question(s))

<b>Q1.</b> Which of the following is a characteristic of a lexical error?	<b>Marks CO BL</b>
<input type="radio"/> It occurs during the syntax analysis phase. <input type="radio"/> It refers to incorrect grammar in the source code.	<input checked="" type="radio"/> It occurs when a token is identified but not found in the lexicon. <input type="radio"/> It occurs during code generation.
<b>Q2.</b> Which of the following represents a sequence of characters that matches a specific pattern?	<b>1 1 1</b>
<input type="radio"/> Token <input type="radio"/> Grammar	<input checked="" type="radio"/> Lexeme <input type="radio"/> Syntax

  

<b>Q3.</b> The grammar $A \rightarrow AA \mid (A) \mid e$ is not suitable for predictive - parsing because the grammar is?	<b>1 2 2</b>
--	--------------

<b>Rubric</b>	<b>Marks</b>
Old Answer is- Left Recursive and the updated answer is option 1. Ambiguous	1

<b>Q4.</b> Which of the following suffices to convert an arbitrary CFG to an LL(1) grammar?	<b>1 2 2</b>
<input type="radio"/> Removing left recursion only <input type="radio"/> Factoring & left recursion removal	<input type="radio"/> Factoring the grammar alone <input checked="" type="radio"/> None of the mentioned
<b>Q5.</b> Assume that the SLR parser for a grammar G has n1 states and the LALR parser for G has n2 states.	<b>1 2 1</b>
<input type="radio"/> n1 is necessarily less than n2 <input type="radio"/> n1 is necessarily greater than n2	<input checked="" type="radio"/> n1 is necessarily equal to n2 <input type="radio"/> None of the mentioned
<b>Q6.</b> An LR-parser can detect a syntactic error as soon as-	<b>1 2 1</b>
<input type="radio"/> The parsing starts <input type="radio"/> It is possible to do so a right-to-left scan of the input.	<input checked="" type="radio"/> It is possible to do so a left-to-right scan of the input. <input type="radio"/> Parsing ends
<b>Q7.</b> In TAC, what does each instruction typically contain?	<b>1 3 1</b>
<input type="radio"/> Two operands and a result <input type="radio"/> One operator and two operands	<input type="radio"/> Three operands and a result <input checked="" type="radio"/> An operator, two operands, and a result
<b>Q8.</b> Which of the following error is expected to recognize by semantic analyzer?	<b>1 3 1</b>
<input type="radio"/> Type mismatch <input type="radio"/> Reserved identifier misuse	<input type="radio"/> Undeclared variable <input checked="" type="radio"/> All of the above

- Q9.** Substitution of values for names whose values are constant, is done in-
- Local optimization
  - Constant folding
  - Loop optimization
  - None of the above
- Q10.** A fragment of code that resides in the loop and computes the same value at each iteration is called a-
- Induction analysis
  - Strength reduction
  - Loop-invariant code
  - None of the above

### Section 2 (Answer all question(s))

- Q11.** How does a symbol table help in compilation? Marks CO BL  
2 1 2

<b>Rubric</b>	<b>Marks</b>
Explanation	2

- Q12.** Explain the need for an input buffer in lexical analysis. 3 1 2

<b>Rubric</b>	<b>Marks</b>
For Explanation	3

- Q13. (a)** Explain the phases of a compiler with a neat diagram. 5 1 2

<b>Rubric</b>	<b>Marks</b>
For each phase (6 phases) explanation 0.5 marks should be awarded.	3
Neat diagram with proper output of each phase.	2

**(OR)**

- (b)** Define the pass structure of a compiler. Explain the types of the pass structure.

<b>Rubric</b>	<b>Marks</b>
Definition of Pass Structure	1
Types of Pass structure (For two types 2 marks for each)	4

### Section 3 (Answer all question(s))

- Q14.** Do left factoring in the following grammar- Marks CO BL  
2 2 3  
 $A \rightarrow aAB / aBc / aAc$

<b>Rubric</b>	<b>Marks</b>
for correct solution	2

**Q15. (a)** Define ambiguous grammar with an example.

8 2 3

Calculate the first and follow functions for the given grammar-

$S \rightarrow A$

$A \rightarrow aB / Ad$

$B \rightarrow b$

Rubric	Marks
Definition of Ambiguous grammar	1
Example	3
First Function (each of 1 mark)	2
Follow Function (each of 1 mark)	2

**(OR)**

- (b)** What is the difference between top-down and bottom-up parsing? Consider the given grammar and check whether the given grammar is LL(1) or not.

$S \rightarrow AB | eDa$

$A \rightarrow ab|c$

$B \rightarrow dC$

$C \rightarrow eC | \epsilon$

$D \rightarrow fD | \epsilon$

Rubric	Marks
Differences between Top down and Bottom up Parsing	2
Calculation of First and Follow function	2
Construction of Parsing Table	3
Answer in Yes or No for checking of Grammar (LL1)	1

#### Section 4 (Answer all question(s))

Marks CO BL

- Q16.** Explain operator precedence grammar. Consider the grammar and construct the operator precedence parser.

$E \rightarrow E+ T / T$

$T \rightarrow T^* F / F$

$F \rightarrow id$

Rubric	Marks
Definition of Operator Precedence Grammar	2
Parsing table of operator grammar	2

**Q17. (a)** Consider the given grammar and construct SLR parser.

$$S \rightarrow L = R \mid R$$

$$L \rightarrow * R \mid id$$

$$R \rightarrow L$$

Rubric	Marks
Data Flow Diagram( LR 0 canonical item sets)	3
Parsing Table	3

(OR)

**(b)** Consider the given grammar and construct CLR parser.

$$X \rightarrow aYd \mid bZd \mid aZe \mid bYe$$

$$Y \rightarrow c$$

$$Z \rightarrow c$$

Rubric	Marks
Data Flow Diagram( LR 1 canonical item sets)	3
Parsing Table	3

### Section 5 (Answer all question(s))

Marks CO BL

4 3 2

**Q18.** Explain SDT with example. How are S attributed SDT different from L attributed SDT?

Rubric	Marks
Definition of SDT	1
Example Of SDT	1
Differences between S attributed and L attributed SDT	2

**Q19. (a)** Define three-address codes. Convert given infix expression  $7+5*3$  into postfix expression by using given SDT.

6 4 3

$$E \rightarrow E + T \quad \{ \text{printf}(“+”) } \}$$

$$E \rightarrow T \quad \{ \}$$

$$T \rightarrow T * F \quad \{ \text{printf}(“*”) } \}$$

$$T \rightarrow F \quad \{ \}$$

$$F \rightarrow \text{num} \quad \{ \text{printf(num.lval) } \}$$

Rubric	Marks
Definition of TAC	2
Tree Diagram of SDT	3
Converted Expression ( Postfix)	1

(OR)

**(b)** Write Three Address Code for the following expression and compute quadruple, triple & indirect triples.

$$-(a \times b) + (c + d) - (a + b + c + d)$$

Rubric	Marks
Generation of Three address codes	1.5
quadruple table	1.5
Triple table	1.5
Indirect Triple table	1.5

**Section 6 (Answer any 2 question(s))****Marks CO BL****Q20.** Explain the structure of an activation record and its components.

5 4 2

Rubric	Marks
Diagram	1
Explanation of parts	4

**Q21.** Discuss different techniques used for loop optimization in compilers.

5 4 2

Rubric	Marks
For techniques name	1
Explanation at least four example one mark each	4

**Q22.** Explain how global data flow analysis helps in optimizing programs.

5 4 2

Rubric	Marks
explanation	5

\*\*\*\*\*