



Faculty of Engineering / Science

End Semester Examination May 2025

CS3CO44 / BC3CO68 Compiler Design

Programme	:	B.Tech. / B.Sc.	Branch/Specialisation	:	CS
Duration	:	3 hours	Maximum Marks	:	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))

Marks CO BL

Q1. Which of the following is a characteristic of a lexical error? 1 1 1

- | | |
|--|---|
| <input type="radio"/> It occurs during the syntax analysis phase. | <input checked="" type="radio"/> It occurs when a token is identified but not found in the lexicon. |
| <input type="radio"/> It refers to incorrect grammar in the source code. | <input type="radio"/> It occurs during code generation. |

Q2. Which of the following represents a sequence of characters that matches a specific pattern? 1 1 1

- | | |
|-------------------------------|---|
| <input type="radio"/> Token | <input checked="" type="radio"/> Lexeme |
| <input type="radio"/> Grammar | <input type="radio"/> Syntax |

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

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

- | | |
|--|---|
| <input checked="" type="radio"/> Ambiguous | <input type="radio"/> Left recursive |
| <input type="radio"/> Right recursive | <input type="radio"/> An operator grammar |

Q4. Which of the following suffices to convert an arbitrary CFG to an LL(1) grammar? 1 2 2

- | | |
|--|--|
| <input type="radio"/> Removing left recursion only | <input type="radio"/> Factoring the grammar alone |
| <input type="radio"/> Factoring & left recursion removal | <input checked="" type="radio"/> None of the mentioned |

Q5. Assume that the SLR parser for a grammar G has n_1 states and the LALR parser for G has n_2 states. 1 2 1

- | | |
|---|--|
| <input type="radio"/> n_1 is necessarily less than n_2 | <input checked="" type="radio"/> n_1 is necessarily equal to n_2 |
| <input type="radio"/> n_1 is necessarily greater than n_2 | <input type="radio"/> None of the mentioned |

Q6. An LR-parser can detect a syntactic error as soon as- 1 2 1

- | | |
|--|---|
| <input type="radio"/> The parsing starts | <input checked="" type="radio"/> It is possible to do so a left-to-right scan of the input. |
| <input type="radio"/> It is possible to do so a right-to-left scan of the input. | <input type="radio"/> Parsing ends |

Q7. In TAC, what does each instruction typically contain? 1 3 1

- | | |
|---|--|
| <input type="radio"/> Two operands and a result | <input type="radio"/> Three operands and a result |
| <input type="radio"/> One operator and two operands | <input checked="" type="radio"/> An operator, two operands, and a result |

Q8. Which of the following error is expected to recognize by semantic analyzer? 1 3 1

- | | |
|--|---|
| <input type="radio"/> Type mismatch | <input type="radio"/> Undeclared variable |
| <input type="radio"/> Reserved identifier misuse | <input checked="" type="radio"/> All of the above |

- Q9.** Substitution of values for names whose values are constant, is done in- 1 4 1
- ☐ Local optimization
 ☐ Loop optimization
☒ Constant folding
 ☐ 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- 1 4 1
- ☐ Induction analysis
 ☐ Strength reduction
☒ Loop-invariant code
 ☐ None of the above

Section 2 (Answer all question(s))

Marks CO BL

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

Rubric	Marks
Explanation	2

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

Rubric	Marks
For Explanation	3

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

Rubric	Marks
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.

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

Section 3 (Answer all question(s))

Marks CO BL

- Q14.** Do left factoring in the following grammar-
 $A \rightarrow aAB / aBc / aAc$

Rubric	Marks
for correct solution	2

- Q15. (a)** Define ambiguous grammar with an example.
 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.
 $E \rightarrow E + T / T$
 $T \rightarrow T * F / F$
 $F \rightarrow id$

4 2 3

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

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

4 3 2

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

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

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

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

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

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

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
