

DR. BABASAHEB AMBEDKAR TECHNOLOGICAL UNIVERSITY, LONERE

Supplementary Semester Examination - Summer 2024

Course: B. Tech.

Branch: Computer Engineering

Semester: VI

Subject Code & Name: BTCOC601_Y20, Compiler Design

Max Marks: 60

Date: 13 / 06 / 2024

Duration: 3 Hrs.

Instructions to the Students:

1. All the questions are compulsory.
2. The level of question/expected answer as per OBE or the Course Outcome (CO) on which the question is based is mentioned in () in front of the question.
3. Use of non-programmable scientific calculators is allowed.
4. Assume suitable data wherever necessary and mention it clearly.

	(Level/CO)	Marks
Q. 1 Solve Any Two of the following.		
A) Define the front-end and back-end of a compiler. Which phases are typically included in each, and what are their primary responsibilities?	Remember Knowledge	6
B) What is lexical analysis? Discuss its role in the compilation process, and explain how it converts the source code into tokens.	Remember Understand	6
C) What are parser generators, such as Yacc or Bison? Explain how they are used to generate parsers and the benefits they provide.	Application Remember	6
Q.2 Solve Any Two of the following.		
A) Discuss techniques for handling lexical errors in a lexical specification. How can a lexical analyzer be designed to report and recover from lexical errors?	Remember Understand	6
B) State and explain the main components of a lexical analyzer generator and their roles in generating a lexical analyzer.	Remember Understand	6
C) Discuss optimization techniques that can be employed by lexical analyzer generators to improve the performance of the generated lexical analyzer.	Understand Apply	
Q. 3 Solve Any Two of the following.		
A) What are parser generators? Describe their role in compiler construction and how they aid in the development of parsers.	Remember Understand	6
B) Discuss the advantages and limitations of using parser generators in compiler design. State the scenarios where using a parser generator would be particularly beneficial.	Knowledge Understand	6
C) What are LR parsers? Describe the different variants of LR parsers, including SLR, LALR, and Canonical LR.	Knowledge Understand	6
Q.4 Solve Any Two of the following.		
A) Describe the purpose and benefits of generating intermediate code in the compilation process. How does intermediate code improve the efficiency and portability of compilers?	Understand Application	6
B) How is intermediate code generated for control flow constructs such as if-else statements and while loops? Give an example of each, showing the source code and the corresponding intermediate code.	Application Understand	6
C) What are the common techniques for optimizing intermediate code? State the examples to illustrate how these optimizations can improve the performance of the generated code.	Understand Apply	6

Q. 5 Solve Any Two of the following.

- | | | | |
|-----------|--|------------------------|----------|
| A) | What is instruction scheduling in the context of code generation? Why is it important for improving the performance of the generated code? | Remember
Understand | 6 |
| B) | Discuss the advantages of using DAGs for representing basic blocks in compiler design. How do DAGs help in optimization and code generation? | Analyze
Understand | 6 |
| C) | What are the common strategies for register allocation, and how do they affect the efficiency of the generated code? | Understand
Apply | 6 |

*** End ***

03.174.255.195 2024-06-13 03:38:17 UTC