

DR. BABASAHEB AMBEDKAR TECHNOLOGICAL UNIVERSITY, LONERE

Bachelor of Technology (Computer Engineering) SEMESTER - 6 Summer 2025 (Regular)

Course :Bachelor of Technology (Computer Engineering) Branch : {FACULTYNAME}

Semester : SEMESTER - 6

Subject Code & Name: BTCOC601 - COMPILER DESIGN

Time : 3 Hours]

[Total Marks : 60

Instructions to the Students:

1. Each question carries 12 marks.
2. Question No. 1 will be compulsory and include objective-type questions.
3. Candidates are required to attempt any four questions from Question No. 2 to Question No.6
4. Use of non-programmable scientific calculators is allowed.
5. Assume suitable data wherever necessary and mention it clearly.

Q1. Objective type questions. (Compulsory Question)

12

1	The output of lexical analyser is:	a. Syntax tree	b. Intermediate code	c. Sequence of tokens	d. Machine code
2	The symbol table is used to store:	a. Only variable names	b. Names and their attributes	c. Assembly codes	d. Tokens
3	What is the action of parsing the source program into proper syntactic classes known as?	a. Interpretation analysis	b. General syntax analysis	c. Syntax analysis	d. Lexical analysis
4	What is the compiler called which runs on one machine and produces code for a different machine?	a. Optimizing compiler	b. One pass compiler	c. Cross compiler	d. Multipass compiler
5	What is the process of finding a parse tree for a string of tokens called?	a. Tokenizing	b. Parsing	c. Analyzing	d. Recognizing
6	What does a top-down parser generate?	a. Rightmost derivation in reverse	b. Rightmost derivation in reverse	c. Leftmost derivation in reverse	d. Leftmost derivation
7	Which of the following software tool is parser generator?	a. Lex	b. Yacc	c. Lex & Yacc	d. None of the mentioned
8	A Synthesized attribute can be calculated by which of the following approaches?	a. postorder traversal approach	b. preorder traversal approach	c. Bottom- Up Approach	d. Top-down Approach
9	When can semantic errors be detected?	a. During runtime	b. During compiler time	c. Both a and b	d. None
10	What is the role of optimizing the compiler?	a. Optimized to take less time for execution	b. Optimize the code	c. Optimize to occupy less space	d. None
11	What is the graph called which shows basic blocks along with their successor relationship?	a. Direct Acyclic graph	b. Control graph	c. Flowgraph	d. Hamiltonian graph

12	Which among the following optimizations is machine independent?			
	a. Peephole optimization	b. Instruction scheduling	c. Loop unrolling	d. Constant folding

Q2. Solve the following.

- A) Draw a neat diagram and explain different phases of compiler. 6
- B) Define compiler. List and explain compiler construction tools. 6

Q3. Solve the following.

- A) Describe the terms: 1. Tokens 2. Lexeme 3. Pattern. 6
- B) Construct NFA using Thompson's construction method $(a+b)^*abb$ 6

Q4. Solve Any Two of the following.

- A) What is left factoring? Left factor the following grammar.
 $S \rightarrow iEtS \mid iEtSeS \mid a$
 $E \rightarrow b$ 6
- B) Consider the following grammar:
 $E \rightarrow E+T \mid T$
 $T \rightarrow T * F \mid F$
 $F \rightarrow (E) \mid id$
Eliminate left recursion and compute first and follow. 6
- C) Show the following Grammar:
 $S \rightarrow AaAb \mid BbBa$
 $A \rightarrow \epsilon$
 $B \rightarrow \epsilon$
Is LL(1) and parse the input string "ba". 6

Q5. Solve Any Two of the following.

- A) Write quadruples, triples and indirect triples for the expression:
 $-(a*b)+(c+d)-(a+b+c+d)$ 6
- B) Explain synthesized attribute and inherited attributes. 6
- C) Consider the following grammar: 6
- $S \rightarrow E$
 $E \rightarrow E1 + T$
 $E \rightarrow T$
 $T \rightarrow T1 * F$
 $T \rightarrow F$
 $F \rightarrow digit$
Write the syntax-directed definition (SDD) for the above grammar.

Q6. Solve Any Two of the following.

- A) What is register descriptor and address descriptor? 6
- What is the cost of the following sequences of instructions?
- i) MOV b,a
ADD c,a
- ii) MOV *R1,*R0
ADD *R2,*R0

- B) Define basic block and flow graph. Give the basic blocks and flow graph for the following sequences of three address code statements. 6
- 1) i=1
 - 2) j=1
 - 3) t1=10*i
 - 4) t2=t1+j
 - 5) t3=8*t2
 - 6) t4=t3-88
 - 7) a[4]=0.0
 - 8) j=j+1
 - 9) if j<=10 goto (3)
 - 10) i=i+1
 - 11) if i<=10 goto(2)
 - 12) i=1
 - 13) t5=i-1
 - 14) t6=88*t5
 - 15) a[t6]=1.0
 - 16) i=i+1
 - 17) if i<=10 goto (13)
- C) Explain the various issues in the design of code generation. 6

051039827

051039827

051039827

051039827

051039827

051039827

051039827

051039827

051039827