

Student name:.....StudentID: .....  
 Course: Compiler Construction CourseID: IT3322E  
 Exam [ ]Midterm [X]final 1 School year:2019-2020 Date:2/1/2020

Mark	Signature of instructor	Signature of examiner

**Read the questions carefully and check exactly ONE option. Write your answer to the answer sheet**

**Question 1:** Consider the context free grammar:

$A \rightarrow Bx \mid yC \mid \varepsilon$ ,  $B \rightarrow CzA$ ,  $C \rightarrow xB$ , where  $\{A, B, C\}$  is the set of nonterminal symbols,  $\{x, y, z\}$  is the set of terminal symbols  $A$  is the start symbol. What is FIRST ( $A$ )?

- A.  $\{y, \varepsilon\}$  C.  $\{y, z, \varepsilon\}$   
 B.  $\{x, y, \varepsilon\}$  D.  $\{x, y, z, \varepsilon\}$

**Question 2:** A parse tree showing the value of attributes at each node is

- A. Annotated Parse Tree  
 B. Attribute Parse Tree  
 C. Semantic Tree  
 D. Syntax Tree

**Question 3:** In which situation, inherited attribute is natural choice:

- A. Evaluation of arithmetic expressions  
 B. Keeping track of variable declarations  
 C. Checking for correct use of L-value and R - value  
 D. None of the above.

**Question 4:** ..... is considered as an instance of a token.

- A. Texeme  
 B. Pattern  
 C. Lexeme  
 D. Mexeme

**Question 5:** Which of the following is a top down parser?

- A. SLR parser  
 B. LALR parser  
 C. Operator precedence parser  
 D. Recursive descent parser.

**Question 6:** A bottom up parser generates

- A. Sequence of productions used in rightmost derivation  
 B. Sequence of productions used in rightmost derivation in reverse  
 C. Sequence of productions used in left most derivation  
 D. Sequence of productions used in left most derivation in reverse

**Question 7:** The lexical analyzer takes ..... as input and produces a stream of .....as output.

- A. Source program, tokens  
 B. Token, source program  
 C. Grammar, Source program  
 D. Regular expression, tokens

**Question 8:**When is the type checking is usually done?

- A. During code optimizaton  
 B. During lexical analysis  
 C. During syntax directed translation  
 D. During syntax analysis

**Question 9:** Which of the following grammars is not LL(1)?

- A.  $S \rightarrow 1SA$ ,  $S \rightarrow 0A1$ ,  $S \rightarrow 2$ ,  $A \rightarrow 0A1$ ,  $A \rightarrow 1$   
 B.  $S \rightarrow aAS \mid b$ ,  $A \rightarrow cS \mid d$ ,  $B \rightarrow c$   
 C.  $S \rightarrow aSa \mid bSb \mid cSc \mid d$   
 D.  $S \rightarrow \varepsilon \mid ab \mid ba \mid aSb \mid bSa$

**Question 10:** The main difference between a sentence and a sentential form is

- A. there is no difference;  
 B. a sentence contains only terminal symbols but a sentential form can contain some non-terminal symbols  
 C. sententialforms are a subset of sentences but the converse is not true  
 D. sentences are derived from S but sentential forms are not

<p><b>Question 11:</b> Which of the following instructions is written in three address code?</p> <p>A. <math>t[i] := x[i+j] + 1</math>  <b>B. <math>t[i] := i</math></b>  C. <math>t[i] := 1 + x[i]</math>  D. <math>a := -t[i]</math></p>
<p><b>Question 12:</b> Program counter in stack calculator is used to?</p> <p>A. store the top of stack address  B. store the base address of stack  C. to store the middle address of the stack  <b>D. to store the next instruction address</b></p>
<p><b>Question 13:</b> The least number of temporary variables required to create a three-address code in static single assignment form for the expression <math>q + r/3 + s - t * 5 + u * v/w</math> is</p> <p>A. 4  <b>B. 8</b>  C. 7  D. 9</p>
<p><b>Question 14:</b> Consider the syntax directed definition shown below.</p> <p><math>S \rightarrow id : = E \quad \{gen(id.place = E.place); \}</math>  <math>E \rightarrow E1 + E2 \quad \{t = newtemp(); gen(t = E1.place + E2.place); E.place = t\}</math>  <math>E \rightarrow id \quad \{E.place = id.place; \}</math></p> <p>Here, <math>gen</math> is a function that generates the output code, and <math>newtemp</math> is a function that returns the name of a new temporary variable on every call. Assume that <math>t_i</math>'s are the temporary variable names generated by <math>newtemp</math>. For the statement '<math>X = Y + Z</math>', the 3-address code sequence generated by this definition is</p> <p>A. <math>X = Y + Z</math>  <b>B. <math>t1 = Y + Z; X = t1</math></b>  C. <math>t1 = Y; t2 = t1 + Z; X = t2</math>  D. <math>t1 = Y; t2 = Z; t3 = t1 + t2; X = t3</math></p>
<p><b>Question 15:</b> Reduction in strength in code optimization means</p> <p>A. replacing run-time computation by compile time computation  B. replacing a costly operation by a relatively cheaper one  <b>C. Both (a) &amp; (b)</b>  D. removing loop invariant computation</p>
<p><b>Question 16:</b> Backus-Naur Form (BNF) is a notation for which of the following:</p> <p><b>A. context-free grammars</b>  B. context-sensitive grammars  C. unrestricted grammars  D. all of the above</p>
<p><b>Question 17:</b> Right parse is</p> <p>A. The sequence of productions used in an arbitrary derivation of a from S.  <b>B. Reversion of the sequence of productions used in rightt derivation of a from S</b>  C. The sequence of productions used in right derivation of a from S  D. None of the above</p>
<p><b>Question 18 :</b> Given grammar <math>S \rightarrow aSb, S \rightarrow c</math> and string <math>aacbb</math>. Which of the following is the next configuration of <math>(q, 2, S1aS1, aSbb\#)</math>?</p> <p>A. <math>(q, 3, S1aS1aS2, cbb\#)</math>  <b>B. <math>(q, 3, S1aS1a, Sbb\#)</math></b>  C. <math>(b, 2, S1aS1, aSbb\#)</math>  D. None of the above</p>
<p><b>Question 19:</b> Left recursion is not permitted for top down parsing <b>and</b> right recursion is not permitted for bottom up parsing</p> <p>A. Yes <b>B. No</b></p>
<p><b>Question 20:</b> Under which of the following circumstances might you choose to implement a programming language using a compiler rather than an interpreter?</p> <p><b>A. Executables for programs in the language should be able to be distributed and executed without the language implementation.</b>  B. Programs in the language need to perform well (run quickly)  C. The language allows the program to generate and execute program code in the language dynamically  D. You would like programmers to be able to detect program flaws statically</p>
<p><b>Question 21:</b> Task of the lexical analysis is</p> <p><b>A. To parse the source program into the basic elements or tokens of the language</b>  B. To build a literal table and an identifier table  C. To build a uniform symbol table  D. All of these</p>
<p><b>Question 22:</b> Consider the grammar shown below</p> <p><math>S \rightarrow i E t S S'   a</math>  <math>S' \rightarrow e S   \epsilon</math>  <math>E \rightarrow b</math></p> <p>In the predictive parse table, M, of this grammar, the entries <math>M[S', e]</math> and <math>M[S', \\$]</math> respectively are</p> <p><b>A. <math>\{S' \rightarrow e S\}</math> and <math>\{S' \rightarrow \epsilon\}</math></b>  B. <math>\{S' \rightarrow e S\}</math> and <math>\{\}</math>  C. <math>\{S' \rightarrow \epsilon\}</math> and <math>\{S' \rightarrow \epsilon\}</math>  D. <math>\{S' \rightarrow e S, S' \rightarrow \epsilon\}</math> and <math>\{S' \rightarrow \epsilon\}</math></p>

<p><b>Question 23:</b> Recursive descent parser is an example of</p> <p>A. Top down backtracking parser  B. Bottom up backtracking parser  <b>C. Predictive parser</b>  D. None of the above</p>
<p><b>Question 24:</b> Function of the storage assignment is</p> <p><b>A. assign storage to all variables referenced in the source program</b>  B. assign storage to all temporary locations that are necessary for intermediate results  C. assign storage to literals, and to ensure that the storage is allocated and appropriate locations are initialized  D. all of these</p>
<p><b>Question 25:</b> How many tokens are there in the following assignment <code>ac := ba (*1.)</code> of KPL?</p> <p>A. 6  B. 7  C. 8  <b>D. None of the above</b></p>
<p><b>Question 26:</b> Which of the following optimizations can be applied to the following code</p> <pre> (1) prod := 0 (2) i := 1 (3) t1 := 4 * i (4) t2 := a[t1] (5) t3 := 4 * i (6) t4 := b[t3] (7) t5 := t2 * t4 (8) t6 := prod + t5 (9) prod := t6 (10) t7 := i + 1 (11) i := t7 (12) if i &lt;= 20 goto (3) (13) ... </pre> <p>A. Dead code elimination  <b>B. Common Sub-expression Elimination</b>  C. Constant Propagation  D. Partial redundancy elimination</p>
<p><b>Question 27:</b> The graph that shows basic blocks and their successor relationship is called</p> <p>A. Directed Acyclic Graph  <b>B. Control Flow Graph</b>  C. Flowchart  D. Syntax graph</p>
<p><b>Question 28:</b> Grammar <math>E \rightarrow TE', E' \rightarrow +TE' \mid \varepsilon, T \rightarrow FT', T' \rightarrow *FT' \mid \varepsilon, F \rightarrow id \mid (E)</math> is</p> <p>A. Ambiguous  B. Depends on given string  C. ambiguous for certain pair of terminations  <b>D. Unambiguous</b></p>
<p><b>Question 29:</b> What is the value of X printed by the following KPL program ?</p> <pre> program COMPUTE; var X : integer ; procedure FIND ( X: integer) ; begin   X := X*X ; end ; begin   X := 2   FIND ( X );   call writeI ( X ); end </pre> <p><b>A. 2</b>  B. 4  C. 8  D. 16</p>
<p><b>Question 30:</b> Which is not a code optimization strategy?</p> <p>A. Constant folding  B. Copy propagation  C. Dead code elimination  <b>D. Control flow graph</b></p>

**Question 31:** The following object code

L1:

<code of condition>

FJ L2

<code of statement>

J L1

L2:

is generated from which statement?

- A. if <condition> then <statement>
- B. if <condition> then <statement> else <statement>
- C. while <condition> do<statement>
- D. do <statement> while <condition>

**Question 32.** A handle corresponds to the left hand side of a production and can be anywhere in the LR parser stack

- A. Yes
- B. No

#### ANSWER SHEET

QUESTION	ANSWER
1	
2	
3	
4	
5	
6	
7	
8	
9	
10	
11	

QUESTION	ANSWER
12	
13	
14	
15	
16	
17	
18	
19	
20	
21	
22	

QUESTION	ANSWER
23	
24	
25	
26	
27	
28	
29	
30	
31	
32	