A **Reg. No. :** | | | | | | | | | | | | |

Question Paper Code: U5C02

B.E. / B.Tech. DEGREE EXAMINATION, NOV 2023

Fifth Semester

Computer Science and Business Systems

21UCB502- COMPILER DESIGN

(Regulations 2021)

Duration: Three hours Maximum: 100 Marks

Answer ALL Questions

PART A - (10 x 1 = 10 Marks)

1. Arrange the give compilation in the correct order CO1-U

    (a) Linking

    (b) Assembling

    (c) Compiling

    (d) Pre-Processing

    (a) D-C-B-A (b) A-C-B-D (c)D-C-A-B (d) A-B-C-D

2. Tokens are specified by CO1-U

    (a) Regular expressions (b) Algebraic expressions

    (c) Arithmetic expressions (d) Boolean expressions

3. A grammar that produces more than one parse tree for some sentence is called as ----- CO1-U

    (a) Ambiguous (b) Unambiguous (c) Regular. (d) none

4. The grammar s→ L=R leads to the presentation of CO1-U

    (a) Arithmetic expression (b) Left to right expression

    (c) Shift from left to right expression (d) An assignment statement

5. Consider the grammar with the following translation rules and E as the start symbol. CO2-App

E->E1#T{E.value=E1.value*T.value}
E->T{E.value=T.value}
T->T1&F{T.value=T1.value+F.value}
T->F{T.value=F.value}
F ->num {F.value = num.value}

Compute E.value for the root of the parse tree for the expression:2 # 3 & 5 # 6 &4.

(a) 200 (b) 180 (c) 160 (d) 40

6. Syntax Directed Translation is ---- CO1-U

(a) Production with Semantic actions (b) Production with LR(0) items

(c) Production with LR(1) items (d) Production only

7. In Algebraic expression simplification, a = a + 1 can simply be replaced by? CO1-U

(a) a (b) INC a (c) DEC a (d) MUL a

8. Code generator uses ______ function to determine the status of available registers and the location of name values. CO1-U

(a) setReg (b)cinReg (c) pfReg (d) getReg

9. The technique of replacing run time computations during compile time is called------ CO1-U

(a) Constant folding (b) code hosting

(c) peephole optimization (d) invariant computation

10. After the code optimization, statement x=x+0 will be CO1-U

(a) x=x (b) x=0 (c) x=x+0 (d) removed from the code

PART – B (5 x 2= 10Marks)

11. Define Compiler and types of compilers CO1-U

12. Consider the following grammar CO2-App

E→E+T | T

T→T * F | F

F→ (E) | id

obtain Left Recursion for the given grammar.

13. What is the intermediate representation of the statement A or B and not C CO2-App

14. What is register descriptor and address descriptor? CO1-U

15. Draw the DAG for the following basic block CO2-App

a= b + c

b = a – d

c = b + c

d = a - d

PART – C (5 x 16= 80Marks)

16. (a) Illustrate how the following high level language statement is transformed into machine code during the compilation process CO2-App (16)
a = (b+c) * (b+c) *2 with the neat sketch of phases of compiler.

Or

(b) Derive DFA for the given regular expression (a|b)*abb. CO2-App (16)

17. (a) Check the following grammar is SLR(1) or not. CO2-App (16)

S->L=R|R
L->*R | id
R->L

Or

(b) Check whether the following grammar is LL (1) or not. CO2-App (16)

S → L = R| R
L → * R | id
R → L

18. (a) (i) Translate the statement CO2-App (8)

```
        c=0
        do{
            if (a>b) then
                    x++
            else
                    x - -
                    c + +
} while(c<5)
```

(ii) Translate the statement CO2-App (8)

```
switch (ch)
{
Case 1:
      C=a+b;
      Break;
Case 2:
      C=a-b;
      Break;
}
```

Or

(b) (i) Draw the quadruple, triple and indirect triples for the foll statement a = (b*c) + (a+b+c). CO2-App (8)

(ii) Draw the quadruple, triple and indirect triples for the foll statement (a + b) * (c + d) - (a + b + c) CO2-App (8)

19. (a) For the statement x = a / (b + c) – d*(e + f), generate three address code and subsequent target code using the simple code generation algorithm CO2-App (16)

Or

(b) For the statement x= (a + b) – ((c + d) – e)generate three address code and subsequent target code using the simple code generation algorithm CO2-App (16)

20. (a) Explain the principal sources of optimization in detail. CO1-U (16)

Or

(b) What do you mean by data flow analysis? How is it useful for optimization purposes? Explain how the data flow equations for the reaching definition is derived. CO1-U (16)