PART C — (1 × 15 = 15 marks)

16. (a) Consider the following basic block, in which all variables are integers and ** denotes exponentiation

a := x ** 2
b := 3
c := x
d := c * c
e := b * 2
f := a + d
g := e * f

Apply the following optimization techniques to this basic block, in order. Compute the result of each transformation.

(i) Algebraic Simplification (3)

(ii) Copy Propagation (3)

(iii) Constant Folding (3)

(iv) Dead Code Elimination (3)

(v) Common Sub-expression Elimination (3)

Or

(b) Construct LR(l) items for the following grammar and draw the transition diagram representing the transitions among CLR items.

$S \rightarrow CC$

$C \rightarrow cC$

$C \rightarrow d$

Show whether the string "cdcd" is accepted by this grammar or not.

————————

Reg. No. : ☐☐☐☐☐☐☐☐☐☐☐

Question Paper Code : 20423

B.E./B.Tech. DEGREE EXAMINATIONS, APRIL/MAY 2022.

Sixth Semester

Computer Science and Engineering

CS 8602 — COMPILER DESIGN

(Regulations 2017)

Time : Three hours   Maximum : 100 marks

Answer ALL questions.

PART A — (10 × 2 = 20 marks)

1. Programmer A has written a program which needs to be modified very frequently. Which of the two languages Visual Basic (or) C++ can he use for his programming? Justify in two sentences. (Neglect other technical and environmental considerations).

2. List out the phases included in the analysis phase of compiler.

3. Write down the CFG for the set of odd length strings in {a, b}* whose first, middle and last Symbols are same.

4. List out the steps for performing LR parsing.

5. What do you mean by Syntax directed translation?

6. Convert the following statement into three address codes

$x = a + (b * -c) + (d * -e)$

Represent the three address codes by Triples.

7. When do you call a variable to be syntactically live at a point?

8. When does dangling references mean?

9. Define constant folding.

10. Identify and write down the optimizations that could be performed on a Peephole.

PART B — (5 × 13 = 65 marks)

11. (a) Elaborate on the different phases of compiler with a neat sketch. Show the output of each phase of the compiler when the following statement is parsed.
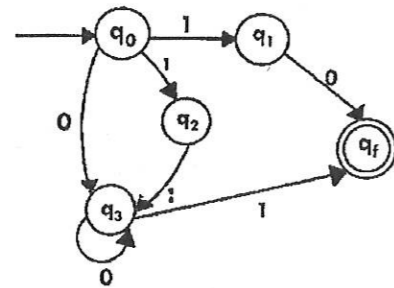
SI = (p *n*r)/100

Where.

n should only be an integer

p and r could be floating point numbers

Or

(b) Convert the following NFA to DFA.

12. (a) Check whether the following grammar can be implemented using Predictive parser. Check whether the string "abfg" is accepted or not using predictive parsing.

$A \rightarrow A$
$A \rightarrow aB \mid Ad$
$B \rightarrow bBC \mid f$
$C \rightarrow g$

Or

(b) Check whether the following grammar can be implemented using Predictive parser. Check whether the string "(a,a)" is accepted or not using predictive parsing.

$S \rightarrow (L) \mid a$
$L \rightarrow L, S \mid S$

13. (a) Write the syntax directed translation for the following code:

$E \mapsto E_1$ or E_2

$E \mapsto E_1$ and E_2

$E \mapsto$ not E_1

$E \mapsto (E_1)$

$E \mapsto id_1$ relop id_2

$E \mapsto$ true

$E \mapsto$ false

Or

(b) Write the syntax directed translation for the following piece of code.

while $a < b$
do
    if c < d
    then
        x : = y+z
else
    x :=y−z

14. (a) Elaborate the issues in developing a code generator.

Or

(b) Construct the basic blocks and flowgraph for the following piece of code.

for i from 1 to 10 do
    for j from 1 to 10 do
    a[i, j] = 0.0;
for i from 1 to 10 do
    a[i,i]=1.0;

15. (a) Describe the parameter passing techniques with an example.

Or

(b) Explain the storage allocation techniques with an example.