VIT®

Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)
CHENNAI

Final Assessment Test(FAT) - NOV/DEC 2025

Programme	B.Tech.	Semester	Fall Semester 2025-26
Course Code	BCSE307L	Faculty Name	Prof. Mercy Rajaselvi Beaulah P
Course Title	Compiler Design	Slot	E1+TE1
		Class Nbr	CH2025260100672
Time	3 hours	Max. Marks	100

Instructions To Candidates
- Write only your registration number in the designated box on the question paper. Writing anything elsewhere on the question paper will be considered a violation.

Course Outcomes
CO1: Apply the skills on devising, selecting, and using tools and techniques towards compiler design
CO2: Develop language specifications using context free grammars (CFG).
CO3: Apply the ideas, the techniques, and the knowledge acquired for the purpose of developing software systems.
CO4: Constructing symbol tables and generating intermediate code.
CO5: Obtain insights on compiler optimization and code generation.

Section - I
Answer all Questions (7 × 10 Marks)

01. Construct a DFA for the regular expression **aaaaa*(b | bb | bbb)** using direct method. **[5 + 5 Marks]**
Note: using **nullable(), firstpos(), lastpos()** and **followpos()**.

[10] (CO1/K3)

02. Using phases of compiler translate the given program construct: **[2 + 2 + 2 +2 +2 Marks]**

```
while (i <= 100)
    {
      sum = sum + i * 2;
      i = i + 1;
    }
```

[10] (CO1/K2)

03. a) Construct SLR parse table for the following grammar **[7 Marks]**

$R \rightarrow R / R$
$R \rightarrow RR$
$R \rightarrow R^*$
$R \rightarrow (R)$
$R \rightarrow a$
$R \rightarrow b$

b) Parse the string: (a / b * a) (a * b / b) **[3 Marks]**
Note: Non-terminal = {R} , Terminals = { / , * , (,) , a , b}

[10] (CO2/K3)

04. Consider the basic block **[5 Marks]**

a. a = 10
b. b = 4 * a
c. t1 = i * j
d. c = t1 + b
e. t2 = 15 * a
f. d = t2 * c
g. e = i
h. t3 = e * j
i. t4 = i * a
j. c = t3 + t4

Which of the following optimizations are possible to be carried out with the above basic block:
i) constant propagation

ii) constant folding
iii) common sub-expression elimination
iv) dead code elimination
v) code invariant
Also construct the DAG for the above basic block. **[5 Marks]**

[10] (CO4/K2)

05. Consider grammar and rules given below for array address translation and generating 3 address code for array references:

$E \rightarrow E_1 + E_2$	{ E.addr := newtemp( ); gen(E.addr '=' E1.addr '+' E2.addr);}
$E \rightarrow E_1 * E_2$	{ E.addr := newtemp( ); gen(E.addr '=' E1.addr '*' E2.addr);}
$E \rightarrow id$	{'E.addr = id.lexeme; }
$E \rightarrow L$	{ E.addr := newtemp( ); gen(E.addr '=' L.array.basname '[' L.addr ']'); }
$L \rightarrow id[E]$	{ L.array = id.lexeme; L.type = L.array.typeofelement; L.addr=newtemp(); gen(L.addr '=' E.addr '*' L.type.width);}
$L \rightarrow L_1[E]$	{ L.array = L_1.array; L.type = L_1.type.typeofelement; t = newtemp(); L.addr = newtemp(); gen(t '=' E.addr '*' L.type.width); gen(L.addr '=' L_1.addr '+' t); }

Function newtemp() returns a new temporary name
L.array.basename means name of the array
L.array.typeofelement means type of the element of the array
L.type.width means width of L.type.
Assume size of integer to be 4 bytes, and lower bound of the arrays to be 0
Let A, B and C be 10x5, 5x7, and 10x7 arrays of integers respectively. Let i, j, and k be integers.
Construct an annotated parse tree for the expression C[i][j]+A[i][k]*B[k][j] **[5+5 Marks]**

[10] (CO3/K4)

06. a) Construct the Directed Acyclic Graph (DAG) for the following Three Address Code: **[5 Marks]**

a = a + b
b = b + c
d = b - c
e = b - d
f = a + b
g = a + f

b) When the instructions are independent, is it possible to change the evaluation order during code generation phase? Justify your answer with example. **[5 Marks]**

[10] (CO4/K1)

07. In the following machine code, R1, R2, R3 and R4 represent CPU registers. a and b represent memory locations. First operand is a destination and second operand is a source.

```
LOAD R1, a

LOAD R2, b

MOV R3, R1

SUB R1, R2

MOV R4, R1

MOV R1, R3

ADD R1, R2

STORE a, R4

STORE b, R1
```

a) For the given machine code, draw the data dependency graph. **[4 Marks]**
b) Assume that the MOV instruction consumes 1 clock cycle, ADD/SUB instructions consume 2 clock cycles and LOAD/STORE instructions consume 3 clock cycles. Consider the three machines 1, 2 and 3 with following instance of resources and find the shortest schedule for each machine to achieve maximum parallelism.

Machine 1: one ALU resource and one MEM resource
Machine 2: one ALU resource and one MEM resources
Machine 3: two ALU resources and one MEM resource **[2 + 2 + 2 Marks]**

[10] (CO5/K6)

Section - II
Answer all Questions (2 × 15 Marks)

08. Consider the following grammar with **N = {F, P, D, L, S, B}** and **T = { int , id , (,) , { , } , , , ; , = , if , else , return , relop}** are the set of non-terminals and terminals respectively. Productions of the grammar is

F → int id(P) { L }
P → P , D | D
D → int id
L → L S | S
S → D ; | id = id ; | if (B) S else S | return id ;
B → id relop id

(a) Obtain the FIRST and FOLLOW sets for each non-terminal. **[4 Marks]**
(b) Construct an LL(1) parsing table for the grammar. **[6 Marks]**
(c) Parse the string int id(int id , int id) {return id}; **[5 Marks]**

[15] (CO2/K3)

09.

a) Write the Three Address Code (TAC) for the following C function: **[5 Marks]**

```
int middle(int a, int b, int c)

{
  int r;

  if (a > b)
        if (b > c) r = b;
        else if (a > c)  r = c; else r = a;
  else
        if (a > c) r = a;
        else if (b > c) r = c; else r = b;
  return r;

}
```

b) Construct a basic block and flow graph for the 3-address code obtained in (a) **[4 Marks]**
c) Translate the expression –(a + b) * (c+ d) + (a + b+ c) into quadruple, triple and indirect triple. **[6 Marks]**

[15] (CO3/K2)

BL-Bloom's Taxonomy Levels - (K1-Remembering,K2-Understanding,K3-Applying,K4-Analysing,K5-Evaluating,K6-Creating)

⇔⇔⇔