ETE Practice

PAGE No.
DATE

(1.) Row and column major Array addressing :-

- **Row major :-**
  - ⇒ $a[n_1][n_2]...[n_k]$
  - ⇒ to access $a[i_1][i_2]...[i_k]$; we have to get content of address :-

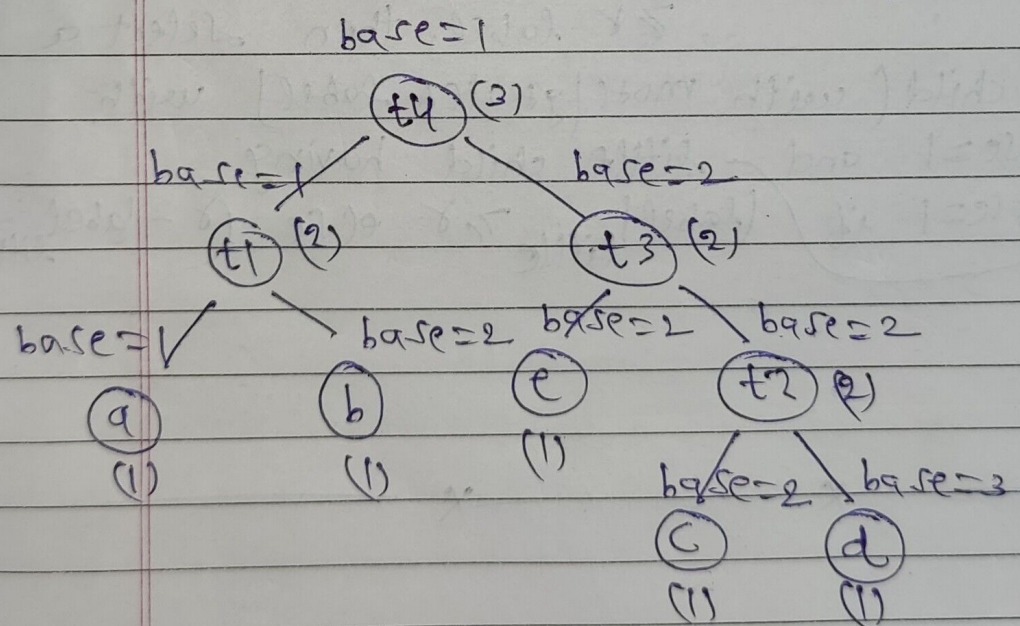
$$\text{(*)}\quad a + \Big((i_1 - low_1) \times n_2 \times n_3 \times \ldots n_k + (i_2 - low_2) \times n_3 \times \ldots \times n_k + \ldots + (i_k - low_k)\Big) \times w$$

  - ⇒ w : width (bytes) in array of each element

- **Column Major :-**

$$\text{(*)}\quad a + \Big((i_1 - low_1) + (i_2 - low_2) \times n_1 + \ldots + (i_k - low_k) \times n_1 \times n_2 \times \ldots n_{k-1}\Big) \times w$$

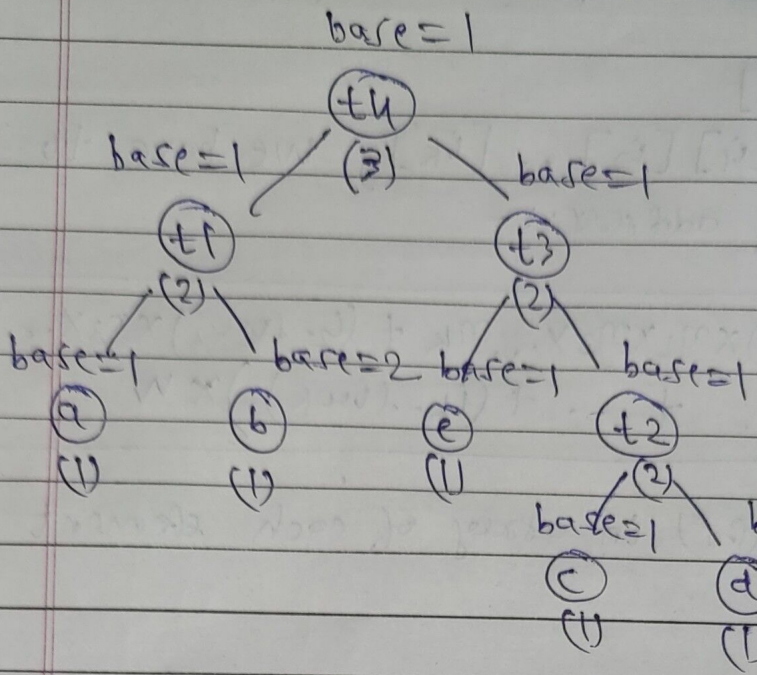
(2.) Ershov Number (sufficient Registers) :-




```
LD R3, a
LD R2, c
Op1 R3, R2, R3
LD R2, e
Op2 R3, R2, R3
LD R2, b
LD R1, a
Op3 R2, R1, R2
Op R3, R2, R3
```

PAGE No.
DATE

(3) Ershov Number (insufficient registers) :- (r=2)




```
LD R2, d
LD R1, c
OP1 R2, R1, R2
LD R1, e
OP2 R2, R1, R2
ST t3, R2
LD R2, b
LD R1, a
OP3 R2, R1, R2
LD R1, t3
OP4 R2, R2, R1
```

∴ If we at node N,

(i) $(Label)_N <= r$ → apply the original algo. for base calculation of left & right node

(ii) $(Label)_N > r$ → If any of them (child) have $\geq r$ label, then select a big child (with most/greater label) with base=1 and little child having base=1 if $(label)_{little} > r$ else $(r - label_{little})$

[In both the cases, result will appear in R8]

PAGE No.
DATE

(4.) SDD for XOR :- (Intermediate code gen)

⇒ $B_1 \wedge B_2 \equiv (!B_1 \text{ and } B_2) \;||\; (B_1 \text{ and } !B_2)$

⇒ $B \rightarrow B_1 \wedge B_2$ { $B_1.true = new\ label();$
$B_1.false = new\ label();$

$B_2.false = B_1.true$
$B_2.true = B.true$

$b_3 = new\ Boolean()$
$b_3.code = B_1.code$
$b_3.false = B.false$
$b_3.true = new\ label()$

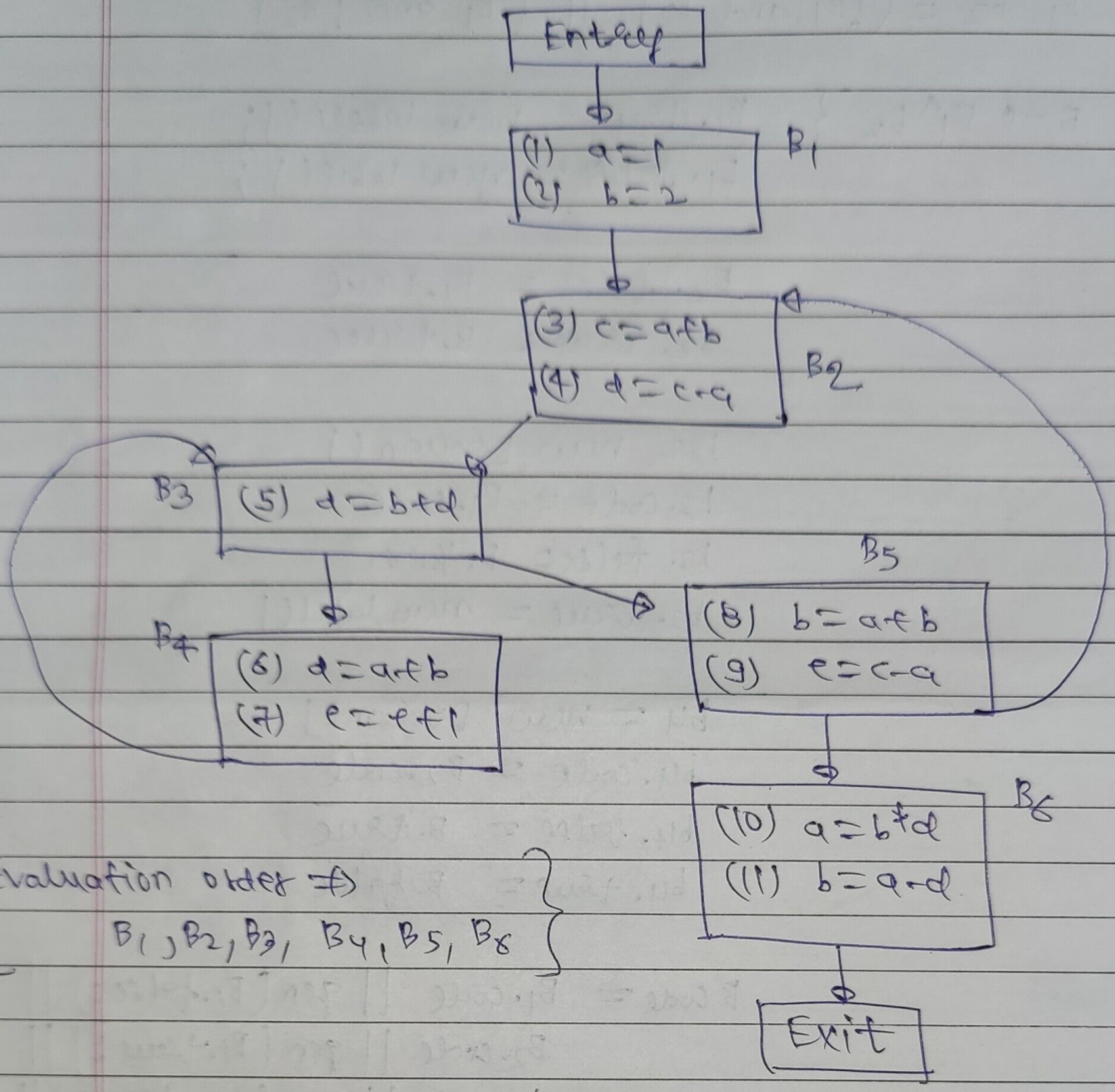
$b_4 = new\ Boolean()$
$b_4.code = B_2.code$
$b_4.false = B.true$
$b_4.true = B.false$

$B.code = B_1.code \;||\; gen(B_1.false:) \;||$
$B_2.code \;||\; gen(B_1.true:) \;||$
$b_3.code \;||\; gen(b_3.true:) \;||$
$b_4.code$

}

PAGE No.
DATE

(5.) Reaching definition :- (Forward Analysis)



Evaluation order ⇒

B1, B2, B3, B4, B5, B6

Block	kill	Gen	In	out	In	out
B1	~~8~~, ~~10~~, 11	1,2	.....		.....	
B2	5,6	3,4	.....		.....	
B3	4,6	5	.....		.....	
B4	4,5,9	6,7	.....		.....	
B5	2,7,11	8,9	.....		.....	
B6	1,2,8	10,11	.....		.....	

(6.) Available Expression :- (Forward Analysis)

	e_kill	e_gen
B_1	ABCEF	ϕ
B_2	CEF	AB
B_3	CEF	ϕ
B_4	CDEF	A
B_5	ACDE	B
B_6	ABCE	F

A: a+b
B: c-a
C: b+d
D: e+1
F: a-d

(7) Live variable analysis : Backward analysis

- Evaluation order ⇒

$B_6, B_5, B_4, B_3, B_2, B_1$

	use	def
B_1	ϕ	a, b
B_2	a, b	c, d
B_3	b, d	d
B_4	a, b, e	d, e
B_5	a, b, c	b, e
B_6	b, d	a, b

And