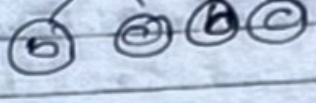
26-08-24
Monday

Day 5 - Analytical Questions

(1) $a = b + c$

$t_1 = a \times a$

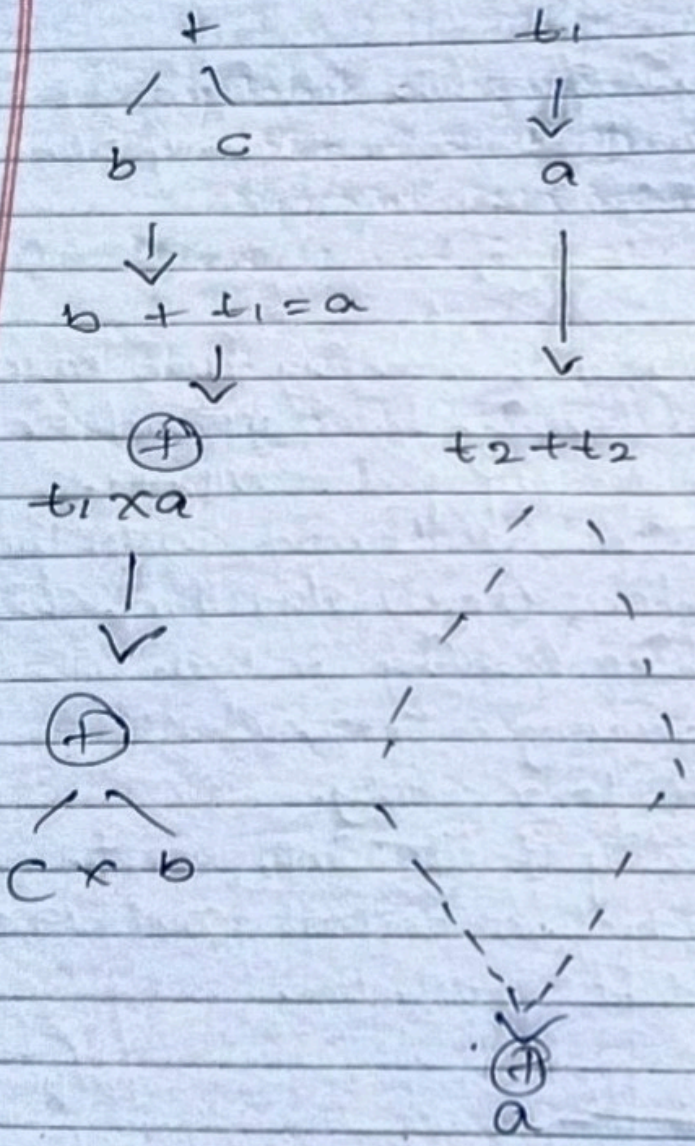


$b = t_1 + a$

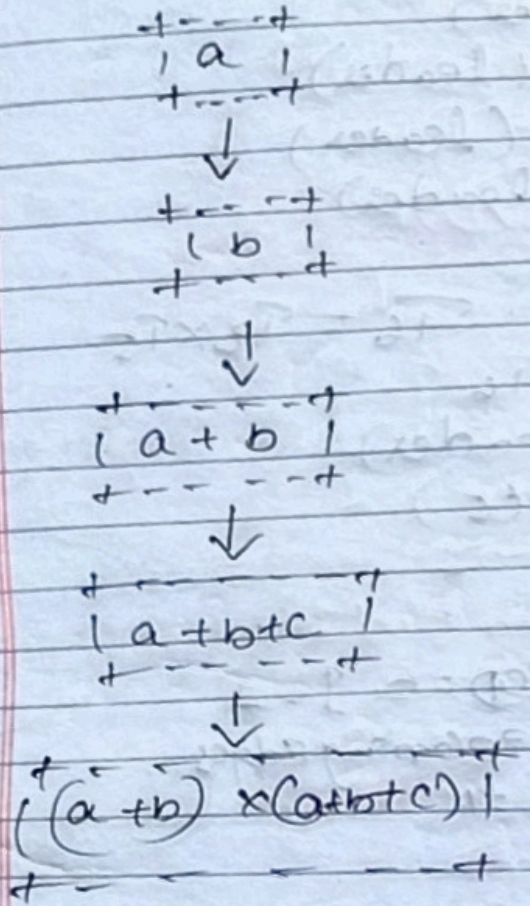
$c = t_1 * b$

$t_2 = c + b$

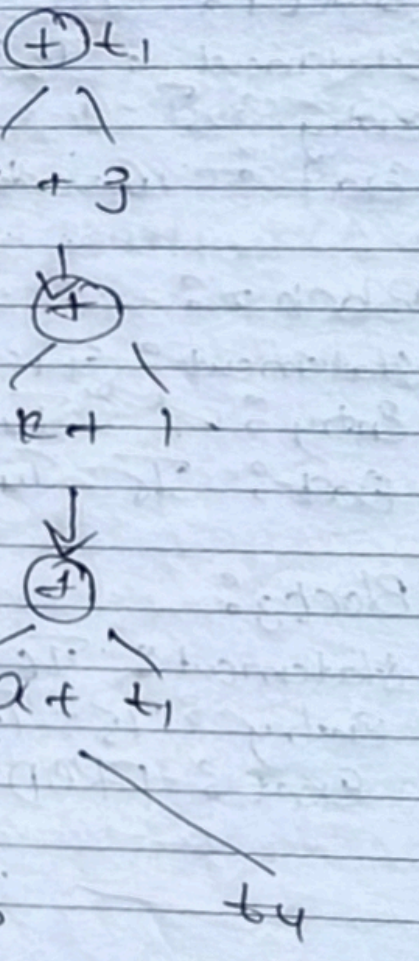
$a = t_2 + t_2$



(2) (a+b) x (a+b+c)



(3) $t_1 = i + j$
$t_2 = k + 1$
$t_3 = a + t_1$
$t_4 = t_2 + t_3$



(4)
PROD = 0 (leader)
I = 1 (leader)
T_2 = addr(A) - 4 (leader)
T_4 = addr(B) - 4 (leader)
T_1 = 4 x I (leader)
T_3 = T_2 [T_1]
T_5 = T_4 [T_1] T_6 = T_3 x T_5
PROD = PROD + T_6
I = I + 1 (leader)
IF I <= 20 GOTO(5)

- Entry Block :
  -> statement : PROD = 0, I = 1
  -> Entry point : NO incoming edges
  -> Exit point : I = 1

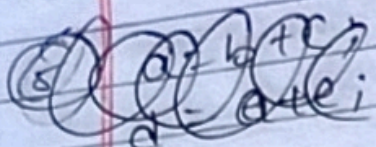
Block 3 :
statement : T_2 = addr(A) - 4 , T_4 = addr(B) - 4
Entry : I = 1
Exit : T_4 = addr(B) - 4

Block 2 :
statement : T_1 = 4 x I , T_3 = T_2 [T_1], T_5 = T_4 [T_1]
Entry : T_4 = addr(B) - 4
Exit : T_5 = T_4 [T_1]

Block 3 :
Statement : T_6 = T_3 x T_5, PROD = PROD + T_6
Entry : T_5 = T_4 [T_1]
Exit : PROD = PROD + T_6

Block 4:
Statement: I = I+1, IF I<=20 GOTO (5)
Entry: PROD = PROD + I6
Exit: IF I<=20 GOTO (5)

Exit Block:
-> Statement: None
-> Entry: If I <= 20 GOTO (6)
-> exit: No outgoing edges.



(12) (i) x = a[i] + 1

```
load R1, a
load R2, i
load R3, 1
Add R2, R2, R1
load x, (R2)
Add x, x, R3
```

(ii) a[i] = b[c[i]]

```
load R1, c
Load R2, i
load R3, (R1+R2)
load R1, b
load R3, i
load R3, (R1+R3)
Store (R2), R3
```

(iii) a[i] = a[i] + b[i]

```
load R1, a
load R2, i
load R3, b
load R4, (R1+R2)
load R5, (R3+R2)
add R4, R4, R5
Store (R1+R2), R4
```

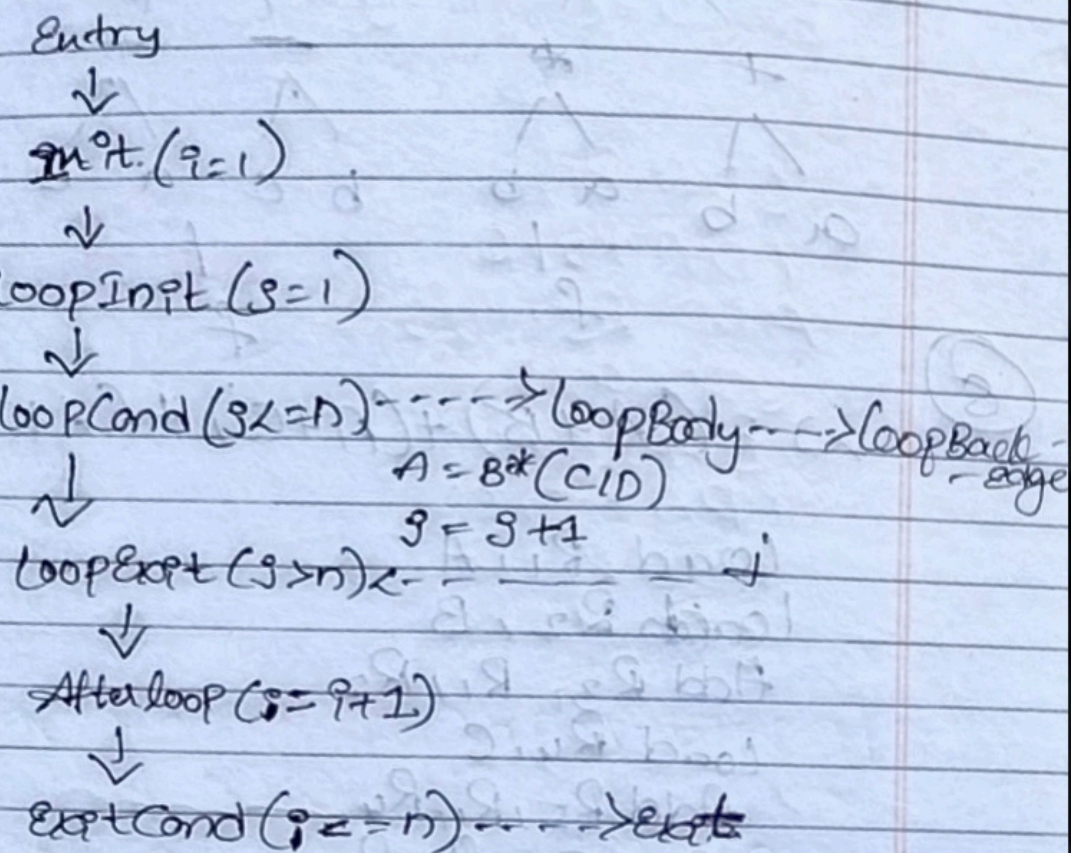
(u) The code is:

```
for(i=1 to n) {
    j = 1;
    while(j<=n){
        A = B*(C/D);
        j = j+1;
    }
}
```

Basic Blocks:

1) Entry Block:
   - Initialize i to 1
2) Loop Initialization Block:
   - Initialize j to 1
3) Loop condition Block:
   - Check if j<=n
4) Loop Body Block:
   - A = B*(C/D)
   - j = j+1
5) Loop Back edge Block:
   - Go back to the loop condition Block.
6) Loop Exit Blocks:
   - Exit the loop if j>n
7) After loop Block:
   - Increment i.
8) Loop Exit condition Block:
   - Check if i<=n
9) Exit Block
   - End of the program.

Flow Graph:

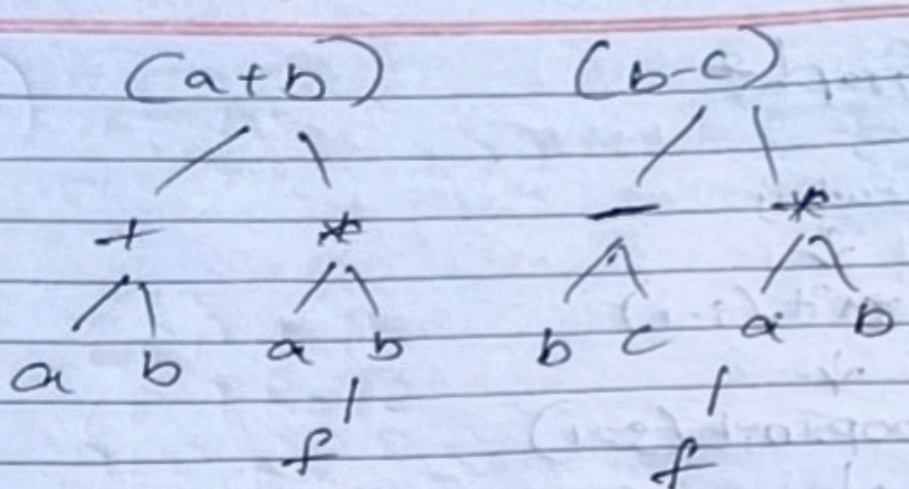


(10) $D = (a-b)*(a-c)+(a-c)$

$t_1 = a-b$
$t_2 = a-c$
$t_3 = t_1 * t_2$
$t_4 = t_3 + t_2$
$D = t_4$

(9) $X = ((a+b)/(b-c) - (a+b)*(b-c)) + f$

$t_1 = a+b$
$t_2 = b-c$
$t_3 = t_1 / t_2$
$t_4 = t_1 * t_2$
$t_5 = t_3 - t_4$
$t_6 = t_5 + f$
$X = t_6$



(8) $w = (A+B)+(A'+C)+(A+C)$

```
Load R1, A
Load R2, B
Add R3, R1, R2
Load R4, C
Add R5, R1, R4
Add R6, R3, R5
Add W, R6, R5
```

(9)

a = b+c	unique expressions are:
c = a+d	'b+c'
d = b+c	'a+d'
c = d-b	'd-b'
a = e+b	'e+b'

→ b+c used in 'a=b+c' & 'd=b+c'
→ a+d used in 'c=a+d'
→ d-b used in 'c=d-b'
→ e+b used in 'a=e+b'

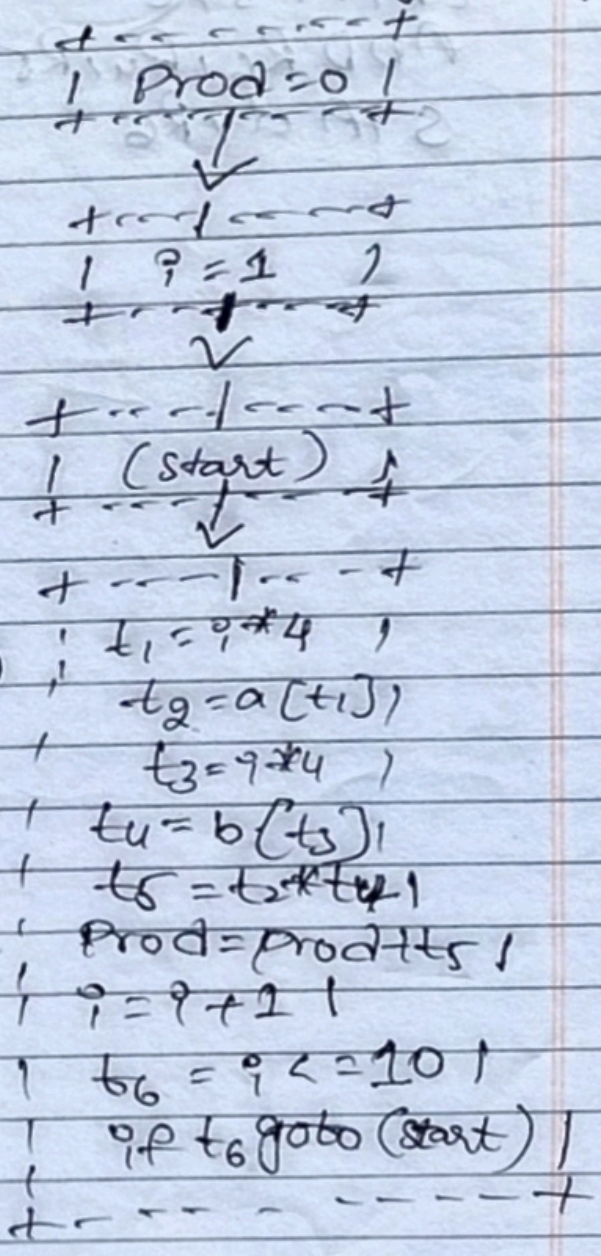
The minimum no. of edges = 4.
The minimum no. of nodes also = 4.

(6)
```
prod=0;
i=1;
do
{
Prod=Prod+a[i]xb[i];
i=i+1;
}while(i<=10);
```

(i) three address code:

```
Prod=0
i=1
(Start)
t1=i*4
t2=a[t1]
t3=i*4
t4=b[t3]
t5=t2*t4
Prod=Prod+t5
i=i+1
t6=i<=10
if t6 goto (start)
```

(ii) Basic Blocks & flowgraphs



(5) a = b+c;
d = a+e; → generate code

LDA R1, b
LDA R2, c
ADD R3, R1, R2
STA a, R3
LDA R4, a
LDA R5, e
ADD R6, R4, R5
STA d, R6