NUMERICAL PROBLEMS

Prob.1. Construct the DAG for the following basic block –

$$t_1 := a + b$$
$$t_2 := c + d$$
$$t_3 := e - t_2$$
$$t_4 := t_1 - t_3$$

and generate code for these three address Statments.

Sol. Consider the following basic block –

$t_1 := a + b$
$t_2 := c + d$
$t_3 := e - t_2$
$t_4 := t_1 - t_3$

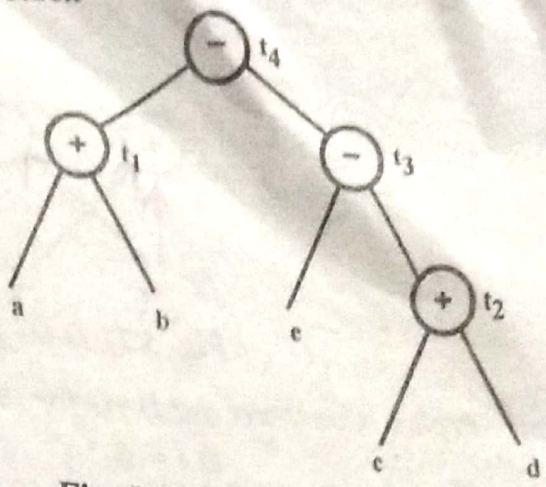

Fig. 5.18 DAG For Basic Block

Code for the three-address statements –

```
MOV  a, R0
ADD  b, R0
MOV  c, R1
ADD  d, R1
MOV  R0, t1
MOV  e, R0
SUB  R1, R0
MOV  t1, R1
SUB  R0, R1
MOV  R1, t4
```

```
MOV  R0, t1
MOV  e, R0
SUB  R1, R0
MOV  t1, R1
SUB  R0, R1
MOV  R1, t4
```

Fig. 5.19 Code Sequence

Prob.2. Construct the DAG for the follwing basic block –

$a := b + c$
$b := a - d$
$c := b + c$
$d := a - d$

Sol.

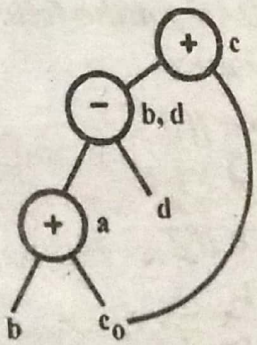

Fig. 5.20 DAG For Basic Block

Prob.3. Construct the DAG for the following basic block –

$a := b + c$
$b := b - d$
$c := c + d$
$e := b + c$

Sol.

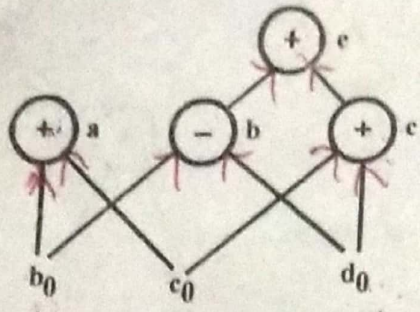

Fig. 5.21 DAG for Basic Block

Prob.4. Construct the DAG for the following basic block –

$$D := B * C$$
$$E := A + B$$
$$B := B * C$$
$$A := E - D$$

(R.G.P.V., Dec. 2003, June 2004, Dec. 2005, June 2007)

Sol.

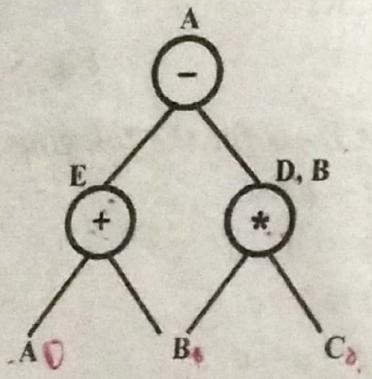

Fig. 5.22 DAG for Basic Block

Prob.5. Construct the DAG for the following basic block –

$$t_1 := 4 * i$$
$$t_2 := a\,[t_1]$$
$$t_3 := 4 * i$$
$$t_4 := b\,[t_3]$$
$$t_5 := t_2 * t_4$$
$$t_6 := Prod + t_5$$
$$Prod := t_6$$
$$t_7 := i + 1$$
$$i := t_7$$
$$If\ i <= 20$$

(R.G.P.V., June 2005, 2006)

Sol. Refer to Q. 2.

Prob.6. Show the DAG for the following statement –

$$Z = X - Y + X * Y * U - V/W + X + V$$

Sol. $Z = X - Y + X * Y * U - V/W + X + V$ ***(R.G.P.V., June 2008)***

The three address code for this is;

$$t_1 = X - Y$$
$$t_2 = t_1 + X$$
$$t_3 = t_2 * y$$
$$t_4 = t_3 * U$$
$$t_5 = t_4 - V$$
$$t_6 = t_5 / w$$
$$t_7 = t_6 + X$$
$$t_8 = t_7 + V$$
$$Z = t_8$$

Fig. 5.23. shows the DAG representation of the given statement as –

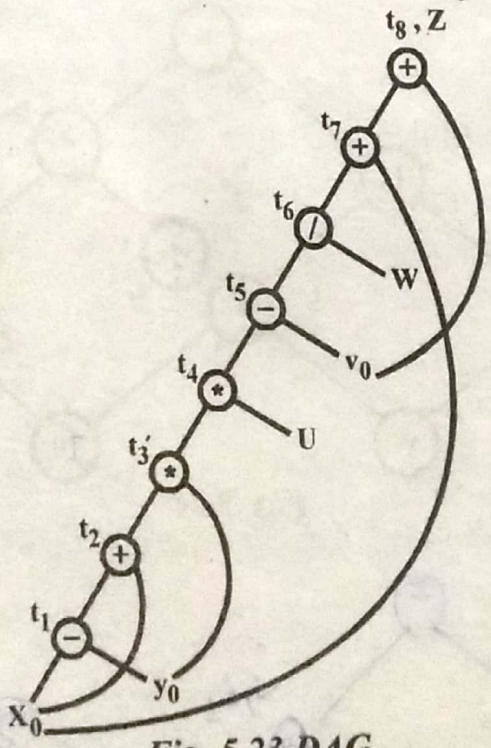

Fig. 5.23 DAG

Prob.7. Construct DAG of basic blocks after converting the code in 3-address representation –

```
i = 1;
j = 2;
repeat
    A[i] = j;
    j = j * 2;
    i = i + 1;
until (i > 10)
```

(R.G.P.V., Dec. 2008)

Sol. The three address code for the given program is as follows –

(i) $i = 1$
(ii) $j = 2$
(iii) $t_1 = \text{width} * i$
(iv) $t_2 = \text{addr}(A) - \text{width}$
(v) $t_2 [t_1] = j$
(vi) $t_3 = j * 2$
(vii) $j = t_3$
(viii) $t_4 = i + 1$
(ix) $i = t_4$
(x) if $(i > 10)$ goto (3)
(xi) Exit,

Construction of DAG of the above written three address code is shown in fig. 5.24 –

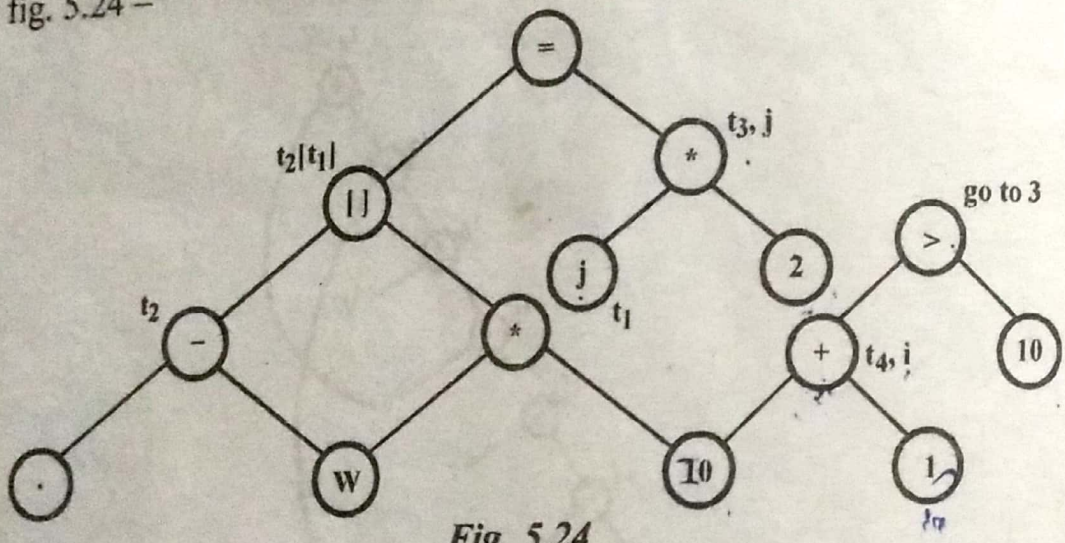

Fig. 5.24