Translating to Three address code.

(1) $c = a + b * c$

TAC

$T_1 = b * c$

$T_2 = a + T_1$

$c = T_2$

(2) $a = (b+c) * (c-d) / (b+c)$

TAC

$T_1 = b + c$

$T_2 = c - d$

$T_3 = T_1 * T_2$

$T_4 = T_3 / T_1$

$a = T_4$

(3) $x = A[y, z]$

Let A be 10 × 20 array
with low1 = low2 = 1

$\therefore n_1 = 10$

$n_2 = 20$

Let $w = 4$

TAC

$t_1 = y \times 20$

$t_1 = t_1 + z$

$t_3 = 4 \times t_1$

$t_4 = A[t_3]$

$x = t_4$

(4) a or b and not c

TAC

$t_1 = not\ c$

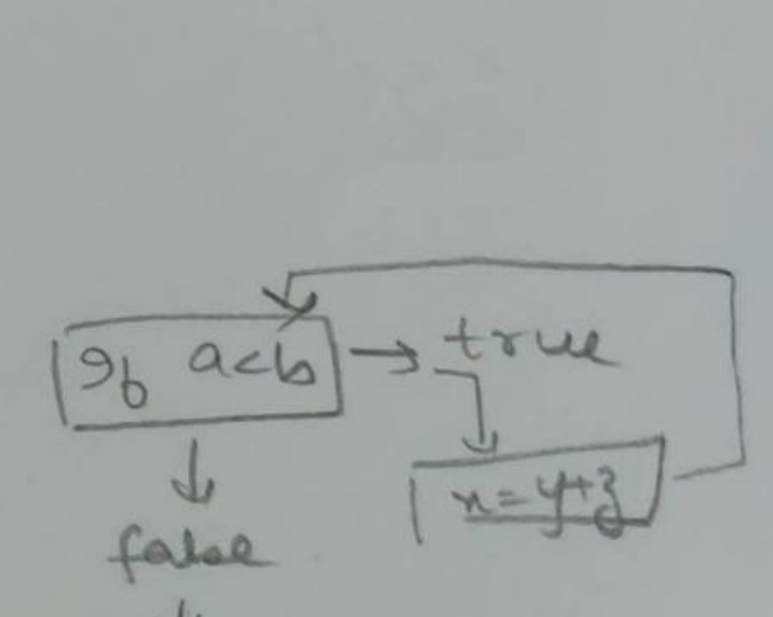
$t_2 = b\ and\ t_1$

$t_3 = a\ or\ t_2$

(5) if a<b then 1 else 0

TAC

```
100   if a<b goto 103
101   t = 0
102   goto 104
103   t = 1
104
```



(6) while a<b do

~~if c<d then~~

x = y + z

end

TAC

```
100   if a<b goto 102
101   goto 104
102   x = y+z
103   goto 100
104
```

⑦ 若 c<d then

$x = y + z$

else

$x = y - z$

TAC

若 c<d goto L1

goto L2

L1 : $t_1 = y + z$

$x = t_1$

go to L3

L2 : $t_2 = y - z$

$x = t_2$

L3 :

```
while a<b do
    if c<d then
        x = y+z
    else
        x = y-z
```

TAC

```
L1: if a<b goto L2
    goto Lnext
L2: if c<d goto L3
    goto L4
L3: t1 = y+z
    x = t1
    goto L1
L4: t2 = y-z
    x = t2
    goto L1
Lnext:
```

(9) Switch E
begin
Case V_1 : S_1
Case V_2 : S_2
:
Case V_{n-1} : S_{n-1}
default : S_n
end

TAC

code to evaluate E into t
goto test

L_1 : code for S1
goto next
L_2 : code for S2
goto next
:
L_{n-1} : code for S_{n-1}
goto next
L_n : code for S_n
goto next

test : If $t = V_1$ goto L_1
If $t = V_2$ goto L_2
:
If $t = V_{n-1}$ goto L_{n-1}
goto L_n
next :

assignment 2

Q1 give three address code of

```
switch (a+b)
    case 0 : x = y+z
    case 2 : x = y+z+2
    case 5 :
            switch (c-d)
            case 1 : x = v+w
            case 2 : x = v-w
            default : x = v*w
default : z = y*z
```