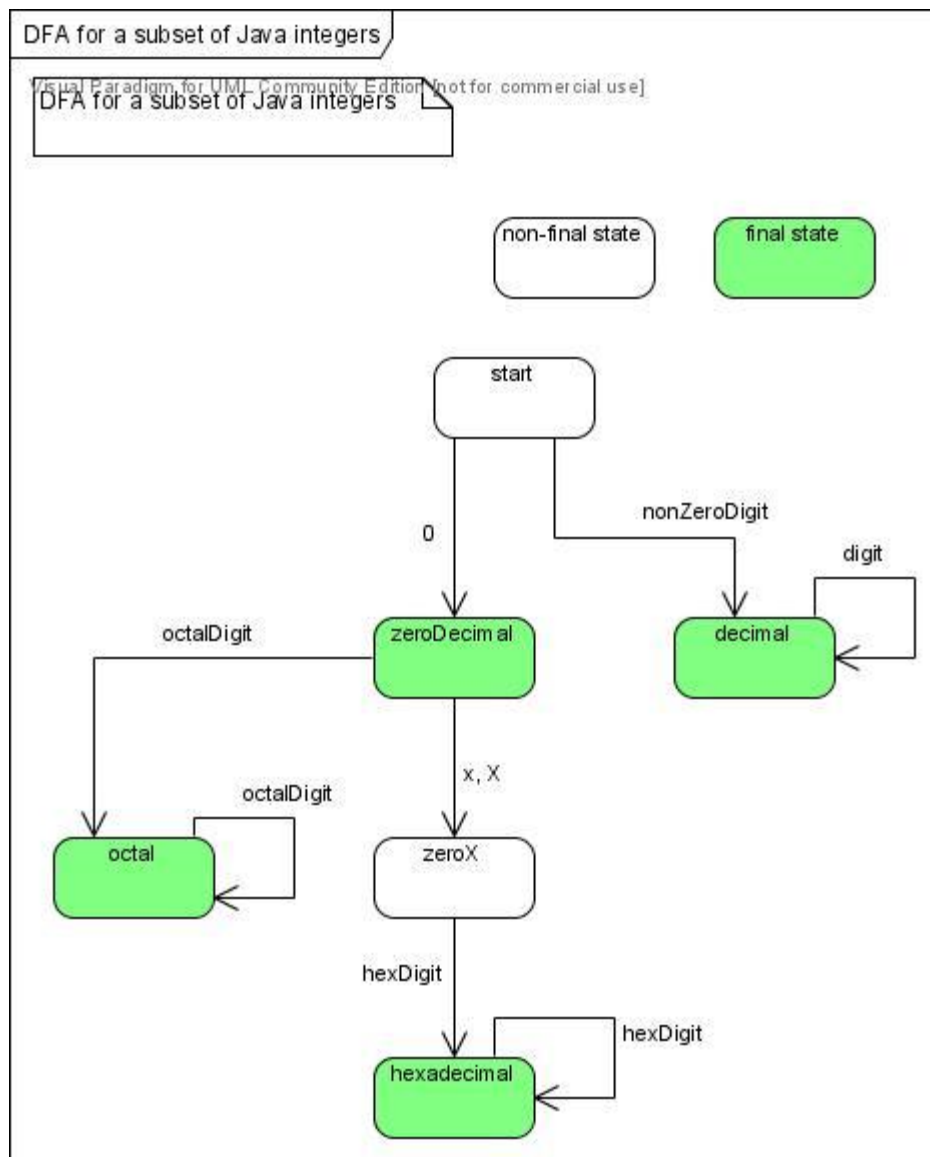


5.



7.

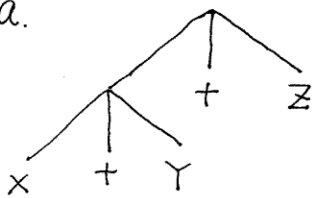
```
void sequence()
{
    if ( t is "(" )
    {
        getToken();
        elements();
        if ( t is ")" )
            getToken();
        else
            print( "Error: ) expected" );
    }
    else
        print( "Error: ( expected" );
}

void elements()
{
    element();
    while ( t is "," )
    {
        getToken();
        element();
    }
}

void element()
{
    if ( t is <id> )
        getToken();
    else
        sequence();
}
```

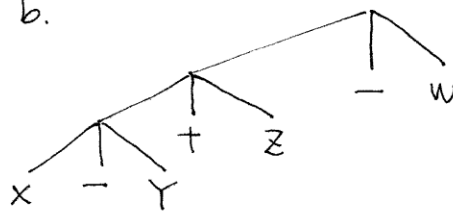
8.2 Since the given production rules are in iterative form, left associativity is used.

a.



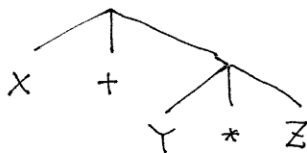
push X
push Y
add
push Z
add

b.



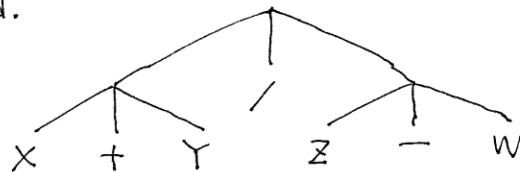
push X
push Y
sub
push Z
add
push W
sub

c.



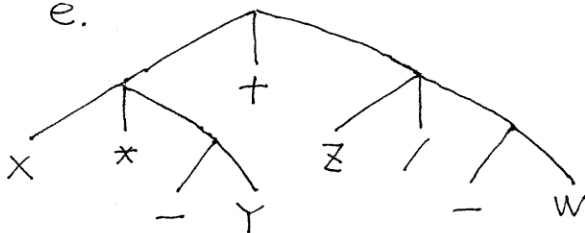
push X
push Y
push Z
mul
add

d.



push X
push Y
add
push Z
push W
sub
div

e.



push X
push Y
neg
mul
push Z
push W
neg
div
add

9.

```
void A()
{
    if ( t is "+" || t is "-" )
    {
        getToken();
        B();
        B();
    }
    else
        print( "Error: + or - expected" );
}

void B()
{
    if ( t is <id> )
        getToken();
    else
        A();
}
```