Iterative Binary Tree Traversal

1. Preorder tree traversal of a binary tree.

```
S <= Empty Stack
S <= (root, 1)
While S != empty do
     (P,i) <= S
     if P != NIL then
          Case i:
               i = 1:
                    S <= (P, 2)
                    Visit node P
               i = 2:
                    S <= (P, 3)
                    S <= (Left(P), 1)
               i = 3:
                    S <= (Right(P),1)
```

2. In-order tree traversal of a binary tree.

```
S <= Empty Stack
S <= (root, 1)
While S != empty do
     (P, i) <= S
     if P != NIL then
          if i = 1 then
               S <= (P, 2)
               S <= (Left(P), 1)
          else
               Visit node P
               S <= (Right(P), 1)
```

3. Post-order tree traversal of a binary tree.

```
S <= Empty Stack
S <= (root, 1)
While S != empty do
     (P,i) <= S
     if P != NIL then
          Case i:
               i = 1:
                    S <= (P, 2)
                    S <= (Left(P), 1)
               i = 2:
                    S <= (P, 3)
                    S <= (Right(P), 1)
               i = 3:
                    Visit node P
```

4. Level order tree traversal of a binary tree.

```
Q <= Empty Queue
Q <= root
While Q != empty do
     P <= Q
     if P != NIL then
          Visit node P
          Q <= Left(P)
          Q <= Right(P)
```