Depth-First Search is an algorithm for traversing or searching a directed graph. It explores as deeply as possible along each branch before backtracking. It can be implemented either using recursion or iterative (a loop without recursion) using a stack.

Here's the iterative version of the algorithm:

1. Initialization:
   Create a stack to store nodes to visit.
   Create a Boolean array called *visited* to track visited nodes (initially all false).
   Choose a starting node.
   Push the starting node onto the stack.

2. Iteration:
   While the stack is not empty:
      Pop a node from the stack
      If the node has not been visited:
         Mark the node as visited.
         Process the node (e.g., print its value).
         For each neighbor of the current node:
            If the neighbor has not been visited, push it onto the stack.

3. Termination:
   The algorithm terminates when the stack is empty, indicating that all reachable nodes have been explored.