DFS using stack

The DFS algorithm:
DFS(G)

```
% initialization
for each $u$ in $V$ do
    $color[u] \leftarrow White$
    $p[u] \leftarrow NIL$
end for
% now the main loop
for each $u$ in $V$ do
    if $color[u] = White$ do
        $DFS - Visit(G, u)$
    end if
end for
```

The procedure DFS-Visit can be implemented recursively, as in the textbook (given last lecture), or can be implemented using stack.

A version of DFS-Visit using stack: DFS-Visit'(G, u):

```
stack $S \leftarrow \varnothing$    % initialize $S$ to the empty stack
$push(S, u)$
while $S$ is not empty do
    $x \leftarrow pop(S)$
    if $color[x] = White$ do
        $time \leftarrow time + 1$
        $s[x] \leftarrow time$
        $color[x] \leftarrow Gray$
        $push(S, x)$
        for each $v$ in $Adj[x]$ do
            if $color[v] = White$ do
                $p[v] \leftarrow x$
                $push(S, v)$
            end if
        end for
    else if $color[x] = Gray$ do
        $time \leftarrow time + 1$
        $f[x] \leftarrow time$
        $color[x] \leftarrow Black$
    end if
end while
```