



✓ DFS 알고리즘 - 재귀

```
DFS_Recursive(G, v)
```

```
    visited[ v ] ← TRUE    // v 방문 설정
```

```
    FOR each all w in adjacency( G, v )
```

```
        IF visited[w] ≠ TRUE
```

```
            DFS_Recursive(G, w)
```



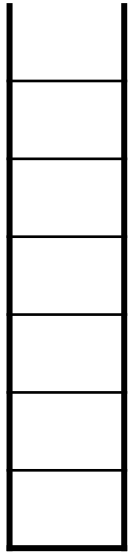
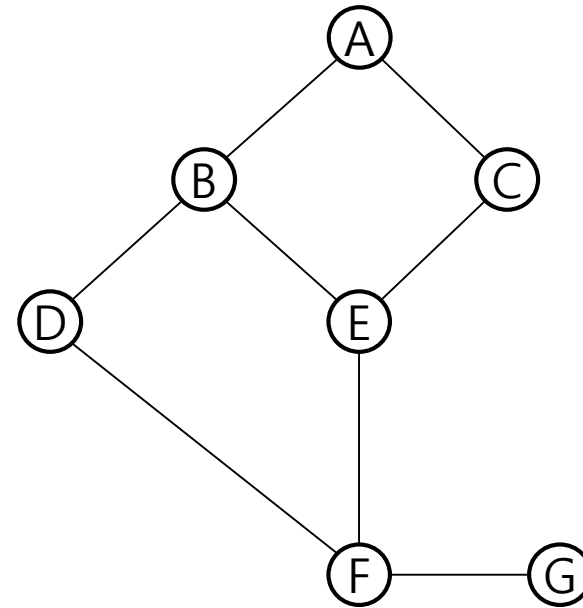
✔ DFS 알고리즘 - 반복

```
STACK s
visited[ ]
DFS(v)
    push( s, v )
    WHILE NOT isEmpty( s )
        v ← pop(s)
        IF NOT visited[v]
            visit( v )
            FOR each w in adjacency( v )
                IF NOT visited[w]
                    push(s, w)
```

DFS 예



```
STACK s
visited[ ]
DFS(v)
  push( s, v )
  WHILE NOT isEmpty( s )
    v ← pop(s)
    IF NOT visited[v]
      visit( v )
      FOR each w in adjacency( v )
        IF NOT visited[w]
          push(s, w)
```



stack

정점	A	B	C	D	E	F	G
	[0]	[1]	[2]	[3]	[4]	[5]	[6]
	F	F	F	F	F	F	F

visited

<연습문제1 – DFS>



- 다음은 연결되어 있는 두 개의 정점 사이의 간선을 순서대로 나열 해 놓은 것이다. 모든 정점을 깊이 우선 탐색하여 화면에 깊이우선탐색 경로를 출력하시오. 시작 정점을 1로 시작하시오.

- 1, 2, 1, 3, 2, 4, 2, 5, 4, 6, 5, 6, 6, 7, 3, 7

- 출력 결과의 예는 다음과 같다.

- 1-2-4-6-5-7-3

- 1-3-7-6-5-2-4

