Depth First Search (DFS)

Difference between DFS and BFS

–exploration of a vertex \( v \) is suspended as soon as a new vertex \( u \) is reached.

–exploration of the new vertex \( u \) begins.

–exploration of \( v \) continues after \( u \) has been explored.

–this can be expressed as a recursive algorithm.
Algorithm $DFS(v : vertex; G : Graph)$

// conduct a breadth first search of $G$,
// starting with vertex $v$
// a vertex $i$ is marked by setting $visited[i] = 1$,
// initially $visited[i] = 0$;

$Visited[v] = 1$;

for all vertices $w$ adjacent to $v$

if ($Visited[w] = 0$)

$DFS(w, G)$;

}
Depth First Search – Example

Stack of visited vertices starting from 1

2
2, 4
2, 4, 8
2, 4, 8, 5
...
2, 4, 8, 6, 3, 7
2, 4, 8, 6, 3
...

Vertices are visited in the order: 1, 2, 4, 8, 5, 6, 3, 7
Exercise: Design a DFS-Span algorithm similar to BFS-Span. DFS starting with vertex \( f \).

Spanning tree?
DFS Spanning Tree Example
Applications

– IP Multicasting

Efficiently transmit (same) data to multiple computers

Multicast requires that there are no loops in the network of computers and routers

Construct a spanning tree with routers and computers as nodes and links between routers as edges

– Web Spiders

Use a “web graph”

A web page is a node. A (directed) edge between two nodes if there is a link in a web page pointing to another.

– Either DFS or BFS can be used to “crawl the web” (Follow the links until no new links can be found.)
Example: “Crawler”: Trace the “crawl” of a DFS based Web Spider for the following web graph.

Exercise: trace the crawl for a BFS based spider.
Summary of Traversal Techniques

Traversal of Graphs

– Breadth First vs. Depth First

– BFS: a node is fully explored before any other can begin.

– DFS: exploration of a node is suspended as soon as another unexplored is found.

– Both are of $O(n)$ time complexity

So which one to use and when?

– BFS is “better” at finding shortest path in a graph

– DFS is “better” at answering connectivity queries (Determining if every pair of vertices can be connected by two disjoint paths)