ID1020 2018-10-05 #12

Directed graphs (Digraphs)
- The edges has direction:

outdegree and indegree
directed path
directed cycle

Can be seen as intersections and one way streets for example.
Has many applications

Problems:
s→t path, shortest s→t path
directed cycle, topological sort
strong connectivity
transitive closure
PageRank

API:
```
Digraph(int V)
Digraph(In in)
addEdge(int v, int w)
Iterable<Integer> adj(int v)
int V()
int E()
Digraph reverse()
String toString()
```

Adjacency lists
- vertex indexed array of lists
- Real world problems tend to be sparse

Reachability problem: which vertices can be reached from v?

DFS (Directed)
To visit a vertex v:
- Mark v as visited
- Recursively visit all unmarked vertices pointing from v.

Every program is a digraph
- Dead-code elimination
- Infinite loop detection

Mark-sweep garbage collection

BFS is same as for undirected graph
- Put s (source) onto FIFO queue, and mark s as visited.
- Repeat until the queue is empty:
- remove the least recently added vertex v
- for each unmarked vertex pointing from v:
  add to queue and mark as visited.

Finds the shortest directed path

Multiple-source shortest path
- Enqueue ←

Topological sort (ex. Precedence scheduling)

DAG - Directed acyclic graph
If cycles exist, topological sort is not possible (⇔)
Run deapth-first search
- Return vertices in reverse postorder

Non-connected vertices may be placed in any order

- First vertex in postorder has outdegree 0.
- second-to-last vertex in postorder can only point to last vertex.
- ...

Proof in presentation

DFS visits each vertex exactly once. The order can be important.
- Preorder
- Postorder
- Reverse postorder

Strongly connected components.
- Two vertices are strongly connected if a directed path exists in both way. This is an equivalence relation.

- A strong component is a maximal subset of strongly-connected vertices.

Kosaraju-Sharir algorithm
Strong components in G is same as in G^R
Kernel DAG: Contract each strong component into a single vertex.

- Compute topological order (reverse postorder) in kernel DAG
- Run DFS, considering vertices in reverse topological order

Kosaraju-Sharir algorithm computes the strong components of a digraph in time proportional to $E+V$.