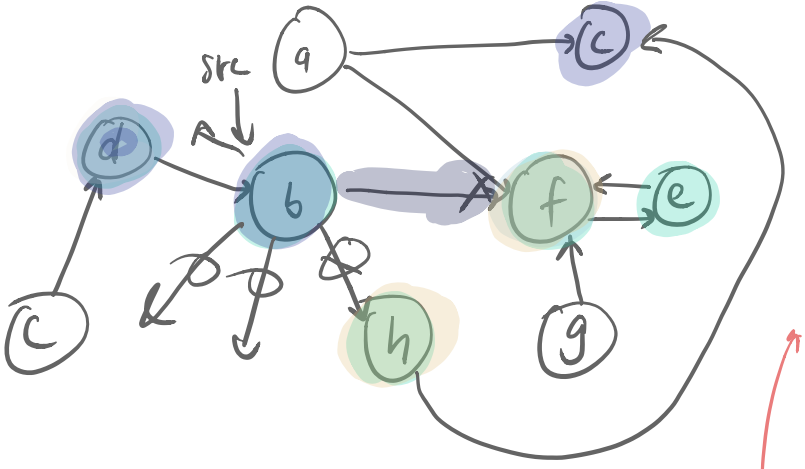
REACHABILITY



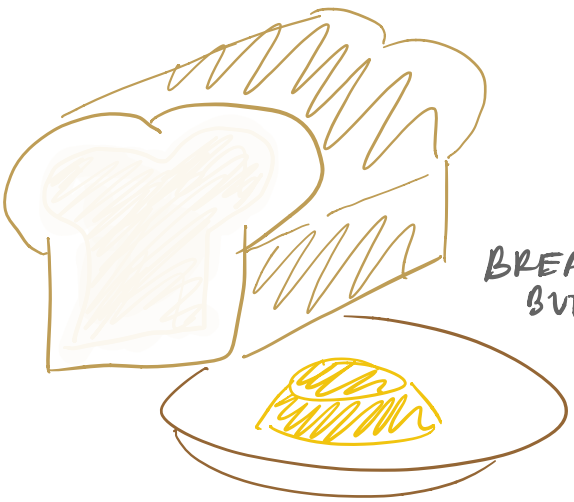
reachable set from a traversal on (d).

{ (d), (b), (h), (f), (e) }

reflexivity ? $x \to x$ ✓
symmetric ? $x \to y \Rightarrow y \to x$ (undirected)
transitive ? $x \to y \wedge y \to z \Rightarrow x \to z$ ✓.

Equivalence relation defined on graphs ! ☺

not assessable.



BREAD and BUTTER of GRAPH ALGORITHMS.

↳ BFS / DFS ☺

Pseudo code : (Recursive DFS)

```
bool reachable (Graph g, Vertex src, Vertex dst)
    Set s = newSet();  // preamble setup.
    return reachableRec(g, src, dst, s)

bool/void reachableRec(Graph g, Vertex src, Vertex dst, Set seen)
    if (src == dst) return 1
    // src ≠ dst : do a step to each of the neighbours of our src.
    addToSet(seen, src)
    for (int i = 0; i < g->nV; i++):
        if (g->edges[src][i] ∧ isInSeen(seen, i) ∧ reachableRec)
            return 1;
```

h
isMember :: Boo)
addToSet

Ok... return Q

What IF we want to print out a path from src → dst?

AUGMENT SET TO BE A PARENT STRUCTURE

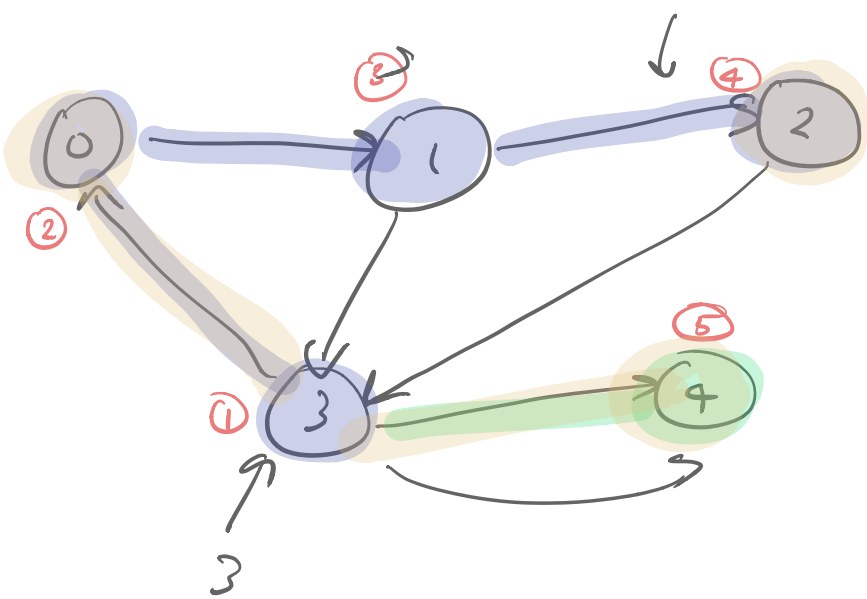
IDEA: Instead of querying if we've seen a vertex.

QUERY for the "PARENT" of a node in our current path.

Q: Do we still need a 'seen' set?

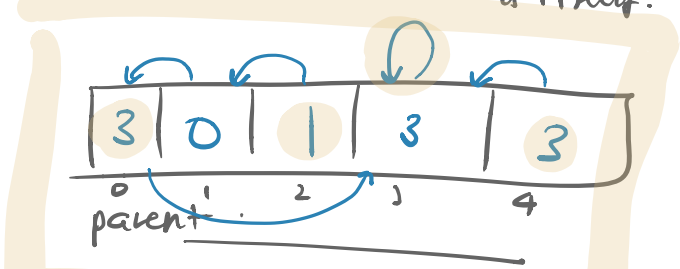
A:

* Below diagram only works for integer labelled vertices.



src = 3

parent of start is itself.

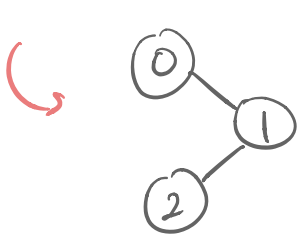


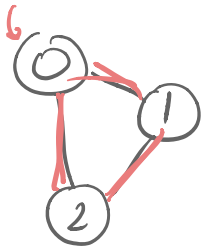
Given a node you want to print a path for, how can you use parent to print a path?

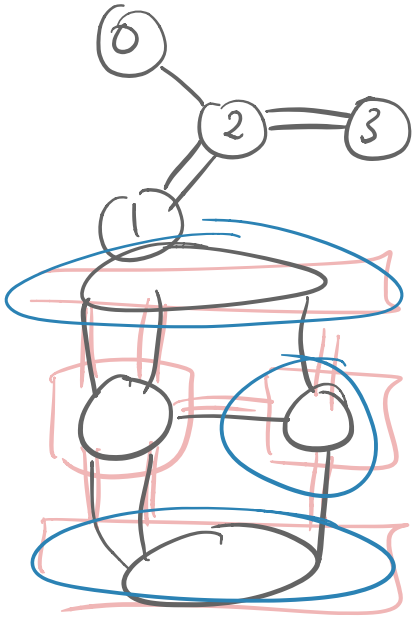
Next lab you'll get to practice this technique!

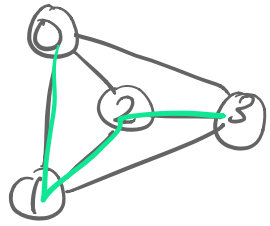
(vertices) (edges)

Hamiltonian / Euler Paths / Circuits









KONIGSBERG PROBLEM?

MAY two vertices with odd degree!