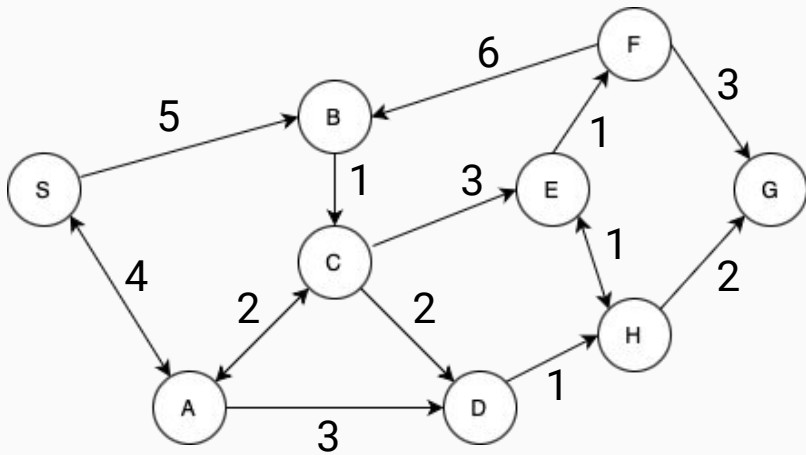


CSE 250 Recitation

October 21~22: Heaps, Dijkstra's Algorithm



Graph Traversal



1. Insert an arbitrary starting node into the [TODO]
2. While the [TODO] is not empty:
 - a. Remove a node from the [TODO]
 - b. Mark the node as visited
 - c. Insert all of the node's unvisited neighbors into the [TODO]

Try the above with TODO as a:

- Queue (BFS)
- PriorityQueue (Dijkstra's)

Compare nodes being marked visited upon removal from TODO vs on add to TODO

Implement PQ with PA1

Give an implementation of a `PriorityQueue` using `SortedList` from PA1

What are the runtimes of the relevant `PriorityQueue` methods with this implementation?

Orderings

We know:

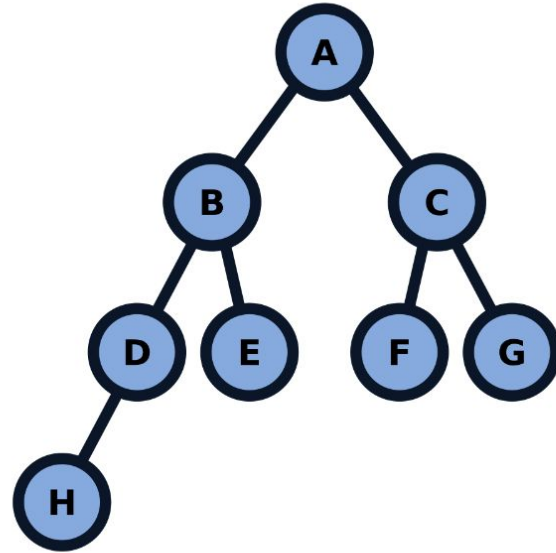
- $A < B$
- $A < C$
- $B < D$
- $B < E$
- $C < F$
- $C < G$
- $D < H$

- What other relationships can we infer?
- What is the smallest number of extra tests we need to...
 - Find the smallest value?
 - Find the second smallest value?
 - Find the third-smallest value?
 - Find the fourth-smallest value?

Heaps

We know:

- $A < B$
- $A < C$
- $B < D$
- $B < E$
- $C < F$
- $C < G$
- $D < H$



Heaps

Are the following arrays valid heaps (min OR max)?

[9 7 4 5 6 2 3]

[20 7 15 2 5 12 9 6 4 1 3]

Heaps

Find tight bounds for inserting sequence of items into a max heap when the sequence is already sorted in **descending order**. What if **ascending order**?

Heapify

Trace the execution of Heapify on the following array (to make a MIN heap)

[9 6 8 1 5 4 15 3 7 14 11 10 2 13 12]

Blooket

<https://dashboard.blooket.com/set/671a80b120a12045e5a5cea0>