

Graph Algorithms / Test Review

Thursday, December 3, 2015 11:40 AM

HW Driver (HWOd)

Within Your lookup method
lookup (val) {
 iter ...
 if (iter.val.equals(val)) {
 return iter.val;
 }
}

Instead
if (iter.val.compareTo(val) == 0) {

Office Hours (SWGN 2D19)

T 1-2 pm

F 11am-12

Final

12:30pm Dec 12th

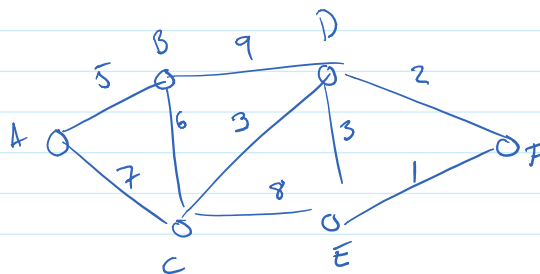
Graph Algorithms

Min-spanning Tree

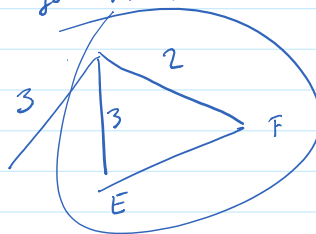
Shortest path tree

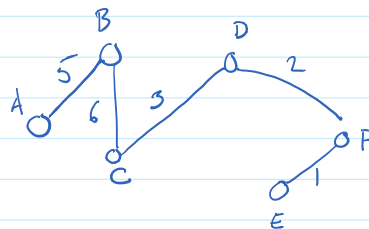
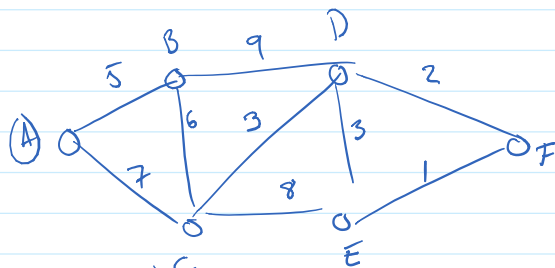
Prim's Algorithm (Also Kruskal's algorithm)

Dijkstra's Algorithm (A* path-finding)



Simple algorithm fails

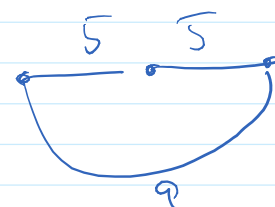




Prim's Algorithm

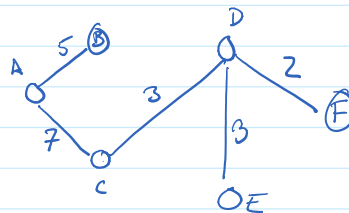
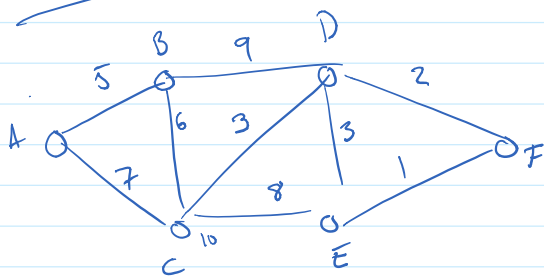
	A	B	C	D	E	F
A	-	5(A)	7(A)	∞	∞	∞
B		-	6(B)	9(B)	∞	∞
C			-	3(C)	8(C)	∞
D				-	3(D)	2(D)
E					-	1(E)
F						-

A, B, C, ... are cities



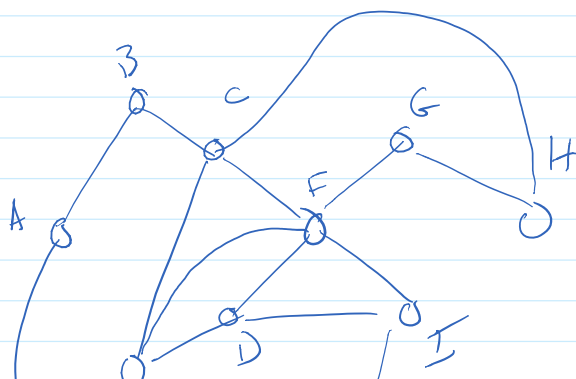
Shortest Path Tree

"Greedy algorithms"



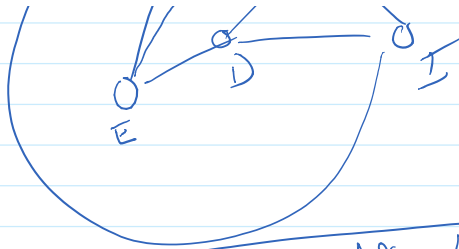
	A	B	C	D	E	F
A	0	5(A)	7(A)	∞	∞	∞
B		0	7(A)	14(B)	∞	∞
C			0	10(C)	15(C)	∞
D				0	3(D)	12(D)
E					0	2(E)
F						0

Both Prim's & Dijkstra's use heaps to maximize efficiency



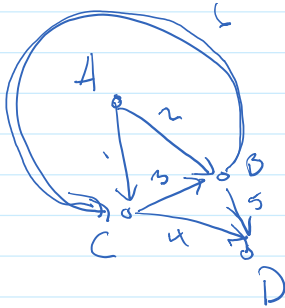
Undirected, Unweighted
Adjacency matrix:

	B	C	D	E	F	G	H
B	1						
C	0	1					
D	0	0	1				
E	0	0	1	1			
F	0	0	1	1	1		
G	0	0	0	0	0	1	
H	0	0	1	0	0	0	1



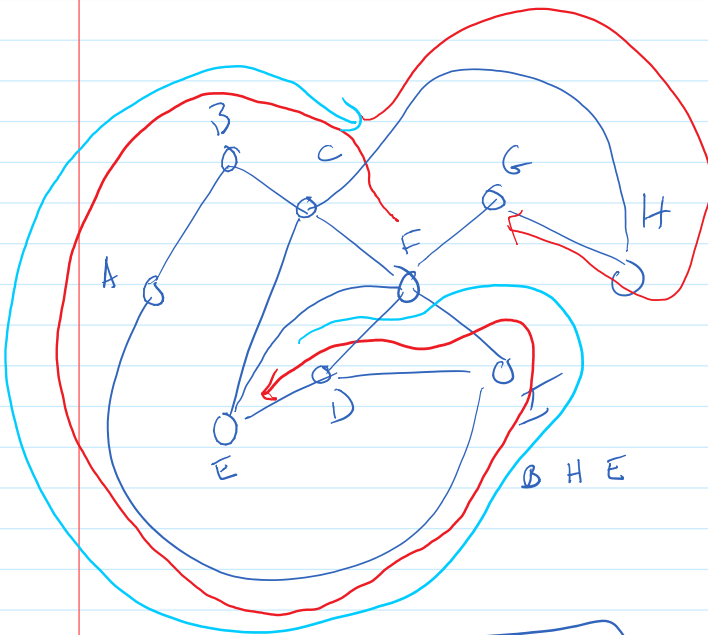
	0	0	0	0	0	1	
G	0	0	0	0	0	1	
H	0	0	1	0	0	0	1
I	1	0	0	1	0	1	0
	A	B	C	D	E	F	G

Adj. List

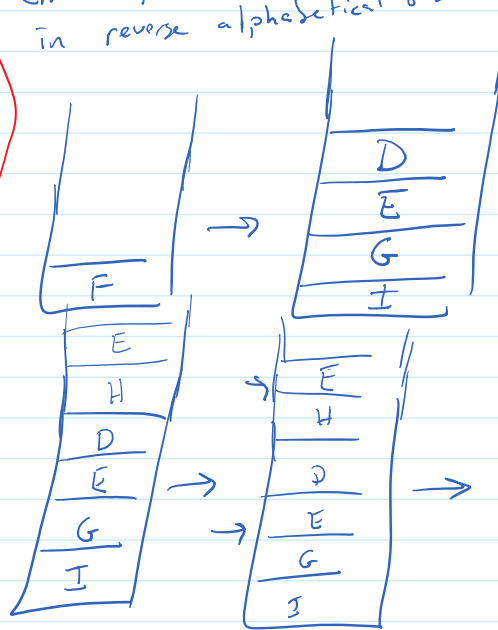


A	→	(C, 1) (B, 2)
B	→	(D, 5) (C, 4)
C	→	(B, 5) (D, 4)
D	→	∅

DFS



DFS on F
entries push onto the stack
in reverse alphabetical order



F C B A I D E H G