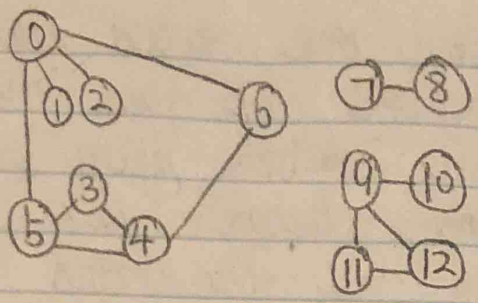
CSC 225 Assignment 4

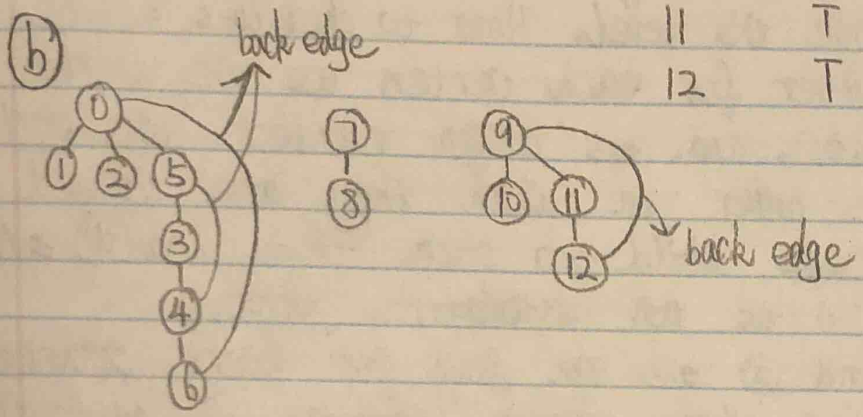
1.



(a)

v	marked[]	EdgeTo[]	id[]
0	T	-	0
1	T	0	0
2	T	0	0
3	T	5	0
4	T	3	0
5	T	0	0
6	T	4	0
7	T	-	1
8	T	7	1
9	T	-	2
10	T	9	2
11	T	9	2
12	T	11	2

(b)

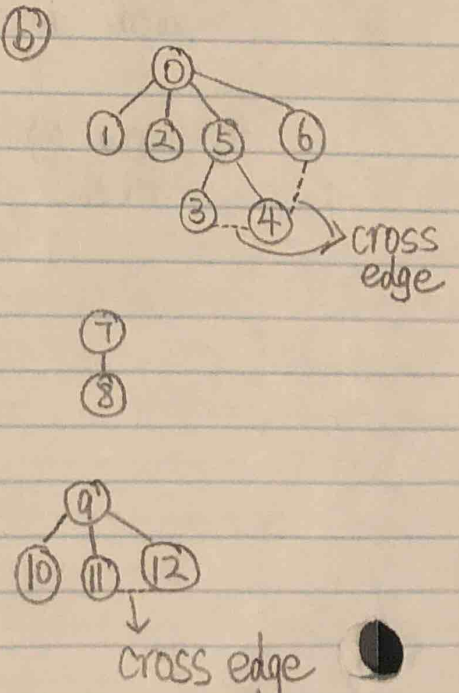


2.

v	marked[]	EdgeTo[]
0	T	-
1	T	0
2	T	0
3	T	5
4	T	5
5	T	0
6	T	0
7	T	-
8	T	7
9	T	-
10	T	9
11	T	9
12	T	9

(a)

(b)



Assignment 4 CSC 225

3. ① if we just want to find one verter that satisfied condition, we can do DFS and just return the very first verter that finished being processed, it's a non-articulation point because it's a leaf.

② if we want to find all the verter that detele it will not disconnect the graph, the idea is using DFS find all articulation points, then the rest of verter are all satisfied condition. How to find articulation points using DFS = ⓐ if the root of DFS tree has two outgoing tree edges, the root is an articulation point. ⓑ for any non-root verter v if it's articulation point, then v's child no back edge to above v's level. How to check this = we find low number for each verter as we talked about in the class, and we assign numbers to each verter in the order in which they are visited by DFS. if v has child w such that $Low(w) \geq Num(v)$, then v is an articulation point.

∴ from ① and ② we can find for every connected graph has a verter whose removal (including all adjacent edges) will not disconnect the graph.

4. Give a bipartite graph G, set size(G) ← ∞, for each verter v in the Graph G = set variable explored ← ∅; distance(v) ← 0;
Toexplore ← {v}; Parent(v) ← null
While Toexplore ≠ 0 =
select verter x from Toexplore;
explored = explored + x; Remove x from Toexplore
for each verter y ∈ (Adj(x) to Parent(x)):
if y ∉ explored =
Parent(y) ← x
distance(y) ← distance(x) + 1
Toexplore ← Toexplore + y
else =
size(G) ← min{size(G), 1 + distance(x) + distance(y)}
return size(G);

Assignment 4 CSC 225

5. ① use topologically sort to sort the DAG
② Assign numbers start at 0 to all vertices on topological sort from left to right.
③ Because in a Hamiltonian path you only can visit each verten once, so if there is an edge between each consecutive pair of vertices in the topological order such as: $(0,1),(1,2),(2,3)\ldots(n-1,n)$, then there exists a Hamiltonian path.