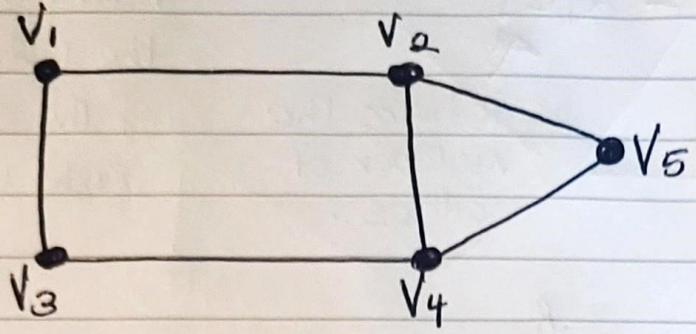
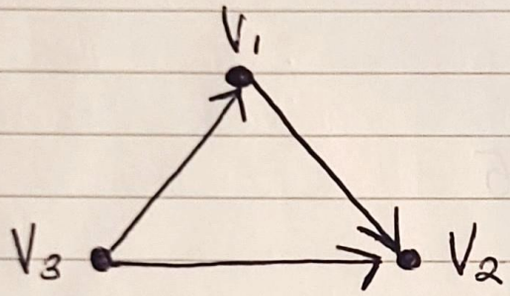
Sample Solutions Tutorial 1 - Graphs

Q1



NOTE: you may have located each vertex in a different position, (so your graph may look different). What matters is that the adjacencies are correct...

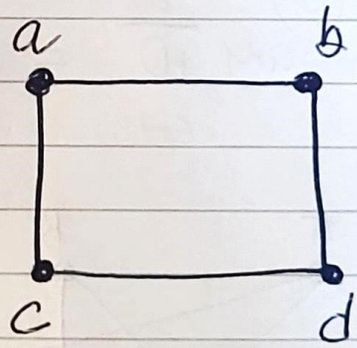
Q2.



NOTE: use arrows for direction

Q3 (a)

	a	b	c	d
a	0	1	1	0
b	1	0	0	1
c	1	0	0	1
d	0	1	1	0



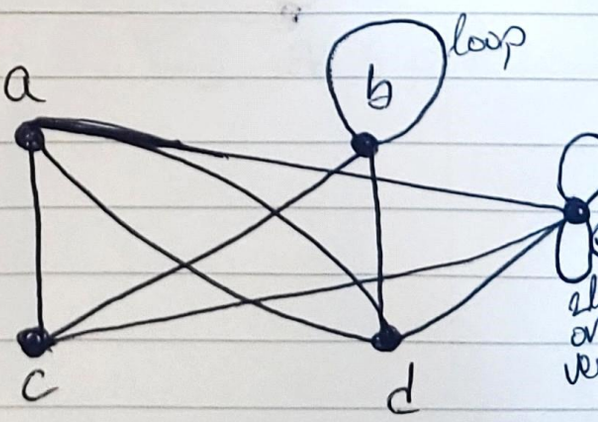
NOTE: each row/col corresponds to a vertex.
DO NOT repeat edges.

(b)

	a	b	c	d	e
a	0	0	1	(2)	1
b (loop)	0	(1)	1	1	0
c	1	1	0	0	1
d	2	1	0	0	1
e	1	0	1	1	(2)

parallel edges

(2) - 2 loops.



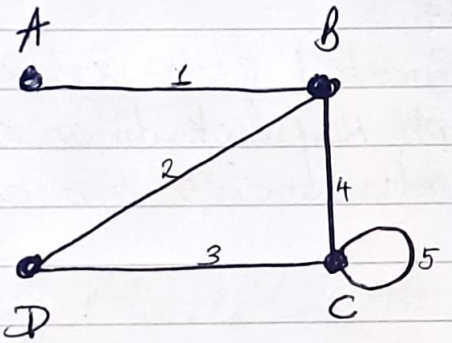
Q4 $\sum \deg(v) = 2|E|$

LHS: sum of the degree of the vertices

RHS: 2 times the number of edges.

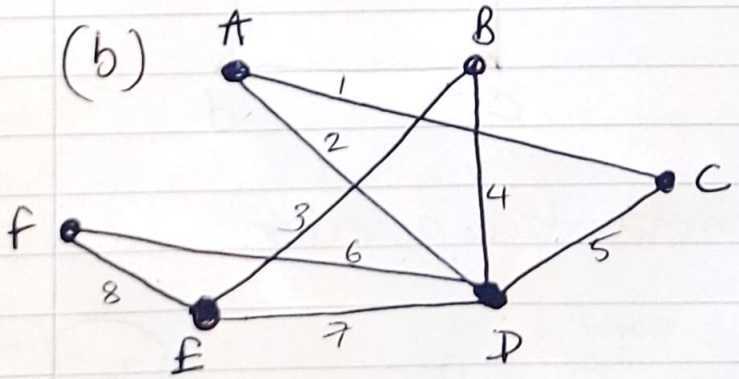
verify → show that the left hand side (LHS) of the eqn = the Right Hand Side (RHS)

(a)



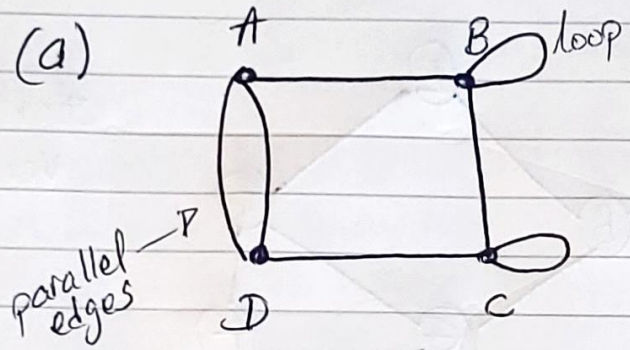
$\deg(A) = 1$ Edges = 5
$\deg(B) = 3$
$\deg(C) = 4$
$\deg(D) = 2$
Sum $10 = 2|5| = 10$
LHS = RHS.

(b)



$\deg(A) = 2$ edges = 8
$\deg(B) = 2$
$\deg(C) = 2$
$\deg(D) = 5$
$\deg(E) = 3$
$\deg(F) = 2$
Sum $16 = 2|8| = 16$
LHS = RHS.

Q5 (a)

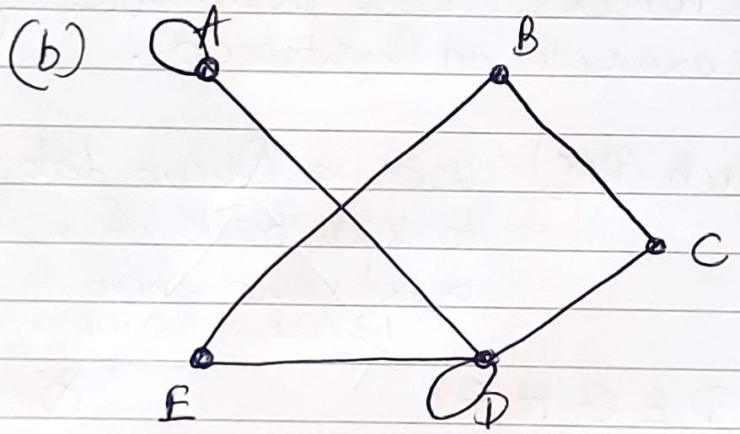


4 vertices ⇒ 4 Rows & 4 Cols

	A	B	C	D	
A	0	1	0	(2)	parallel edges
B	1	(1)	1	0	loop
C	0	1	(1)	1	loop.
D	(2)	0	1	0	

parallel edges.

(b)



	A	B	C	D	E
A	1	0	0	1	0
B	0	0	1	0	1
C	0	1	0	1	0
D	1	0	1	1	1
E	0	1	0	1	0

5 vertices ⇒ 5 Rows & 5 Cols in the resulting Adjacency Matrix.

Q6. (a) Hamiltonian circuit: A a B b C h D j E i F c A

include every vertex of the graph exactly once & first and last vertex coincide (Need not use all edges)

vertices. edges

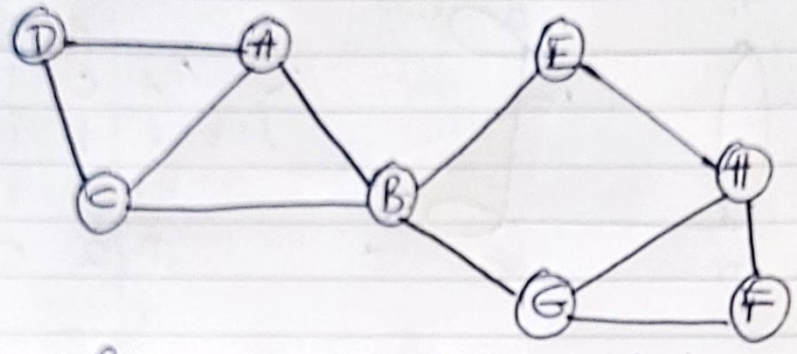
could also use edge d

(b) No Hamiltonian circuit exists: must revisited a vertex...

example Hamiltonian Path: A a G b B c C j D l E g F

additional Hamiltonian paths exist...

Q7



Start at Vertex A

NOTES if there is a decision between multiple neighbour vertices, choose alphabetically.
See video with example on Brightspace.

Breadth first Search (BFS): uses a Queue data structure to keep track of visited vertices & determine where to go next... i.e. look at the top of the queue.

BFS Output: A B C D E G H F

Queue: ~~B~~ ~~C~~ ~~D~~ ~~E~~ ~~G~~ ~~H~~ ~~F~~

NOTE: visit adjacent vertices (alphabetically) before moving to the next phase. When finished with all adjacent vertices, look to the top of the queue to determine where to go next. De-queue each vertex when finished/visited all adj vertices....

→over

Depth first Search (DFS): uses a stack data structure.

DFS Output: A B C D E H F G

NOTE: Move as far away from the starting vertex as possible before back tracking. use the stack to determine where to go next....

~~G~~
~~F~~
~~H~~
~~E~~
~~D~~
~~C~~
~~B~~
Stack: ~~A~~

— Complete the lect 1 QUIZ on Brightspace