CMSC 351 Summer 2025 Homework 7

Due Thursday 7 August 2025 by 23:59 on Gradescope.

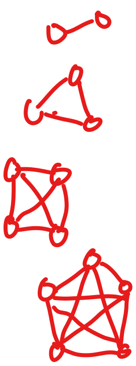
Directions:

- Homework must be done on printouts of these sheets and then scanned properly, or via Latex, or by downloading, writing on the PDF, and uploading. If you use Latex please do not change the Latex formatting.
- Do not use your own blank paper!
- The reason for this is that Gradescope will be following this template to locate the answers to the problems so if your answers are organized differently they will not be recognized.
- Tagging is automatic, you will not be able to manually tag.

---

1. If a simple graph has $V \geq 2$ vertices and $E \geq 0$ edges, explain why $E \leq V^2$. [10 pts]

   Solution:



Find max E for given V

V	E	
2	1	$\frac{2\cdot 3}{2} = 3$
3	3	$\frac{3\cdot 4}{2} = 6$
4	6	$\frac{4\cdot 5}{2} = 10$ ✓
5	10	

formula for max $E = \frac{V(V-1)}{2} \leq V^2$

$\frac{V^2 - V}{2} \leq V^2$ holds true for all $V \geq 2$

$\frac{3^2-3}{2} = \frac{9-3}{2} = \frac{6}{2} = 3$

$3 \leq 9$

$\frac{10^2-10}{2} = \frac{100-10}{2} = \frac{90}{2} = 45 \leq 100$

so at most, $E = \frac{V(V-1)}{2}$ so $E \leq \frac{V(V-1)}{2} \leq V^2$

2. For asymptotic complexity we've only been using one variable but it's fine to have more than one variable inside Θ, $\mathcal{O}$, and Ω. The expression carries over to the definition and each variable gets its own bound. So if V and E are variables then we could have, for example: [10 pts]

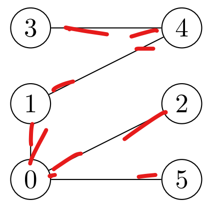
$$f(V,E) = \mathcal{O}(V+E) \text{ if } \exists V_0, E_0, C > 0 \text{ such that } \forall V \geq V_0, \forall E \geq E_0\,,\ f(V,E) \leq C(V+E)$$

   Prove from the definition that $V \lg E = \mathcal{O}(V \lg V)$.

   Solution:

Looking for V lg E = O(V lg V), so we are trying to find an UPPER BOUND.
so we really want V lg E <= V lg V

3. Given the following graph:



(a) Fill in the adjacency matrix for this graph below: [6 pts]

	0	1	2	3	4	5
0	0	1	1	0	0	1
1	1	0	0	0	1	0
2	1	0	0	0	0	0
3	0	0	0	0	1	0
4	0	1	0	1	0	0
5	1	0	0	0	0	0

(b) Write down the adjacency list for this graph. Please put each sublist in increasing order to help us grade more easily: [4 pts]

Solution:

0 1 2 3 4 5

AL = [[1,2,5],[0,4],[0],[4],[1,3],[0]]

4. Justin thinks that if a simple graph has $V \geq 2$ vertices and $E \geq 1$ edges then $V \leq 2E$. His proof goes like this: [5 pts]

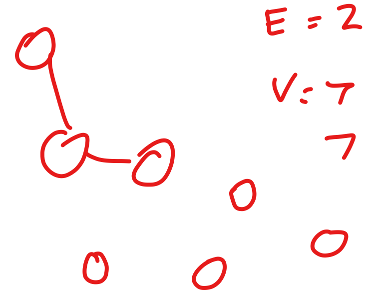
"Each edge requires the existence of two vertices, suggesting $V = 2E$, but vertices are allowed to be shared by edges, meaning we might have fewer vertices."

What is wrong with his proof?

Solution:

- Simple meaning no loops and no multiple edges.
But simple $\neq$ connected

True, each edge requires the existence of 2 vertices but a vertex doesn't require the existence of an edge in a simple graph. if the graph is not connected, that means that there can be any amount of vertices with degree 0 that are technically still part of the graph, so for any simple graph, there is a version of that graph with X>2E unconnected, degree 0 vertices that can increase the vertex count above 2E.



$E = 2$

$V = 7$

$7 \not\leq 2(2)$

5. Suppose the Shortest Path Algorithm is run a graph with nine vertices labeled 0 to 8, starting at the vertex 3. No target is specified so it runs until the queue is empty. The progression of the queue is shown here:

$x=3$	Starting Queue	3 ← $s=3$
	Then	6,7
$x=6$	Then	7,0,2
$x=7$	Then	0,2,1
$x=0$	Then	2,1
$x=2$, $x=1$	Then	1,5,8
	Then	5,8
$x=5$	Then	8,4
$x=8$	Then	4
	Ending Queue	Empty

$Q = 3,$

(a) What will the predecessor list look like? Use N for NONE/NULL. [10 pts]

Index	0	1	2	3	4	5	6	7	8
Entry	6	7	6	N	5	2	3	3	2

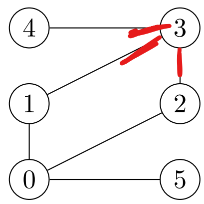
(b) What will the distance list look like? [10 pts]

d[] =

Index	0	1	2	3	4	5	6	7	8
Entry	2	2	2	0	4	3	1	1	3

$d = (\infty, \infty, \infty, 0, \infty, \infty, \infty, \infty, \infty)$
0 1 2 3 4 5 6 7 8

6. Given the following graph and cute little mouse: 1032



Suppose the mouse starts at vertex x and does BFT according to the queue implementation with pushing in increasing numerical order.

$Q = \{ \cancel{0}, \cancel{1}, \underbrace{2, 5, 3, 4} $

(a) For each x give the order in which the vertices will be visited. [12 pts]

		0	1	2	3	4	
4s	$x = 0$	0	1	2	5	$\boxed{3}$	4
2s	$x = 1$	1	0	$\boxed{3}$	2	5	4
2s	$x = 2$	2	0	$\boxed{3}$	1	5	4
0s	$x = 3$	$\boxed{3}$	1	2	4	0	5
1s	$x = 4$	4	$\boxed{3}$	1	2	0	5
4s	$x = 5$	5	0	1	2	$\boxed{3}$	4

$[\cancel{1}, \cancel{0}, \cancel{3}, \underbrace{2\ 5\ 4}]$

$Q = \{ \cancel{2}, \cancel{0}, \cancel{3}, \underbrace{1, 5, 4} \rightarrow$

$Q: \{ \cancel{3}, \cancel{1}, \cancel{2}, \cancel{4}, 0$

$Q = \{ \cancel{4}, \cancel{3}, \cancel{1}, 2, 0, 5$

$Q = \{ \cancel{5}, \cancel{0}, \cancel{1}, \cancel{2}, \cancel{3}, 4$

(b) Suppose that, starting at the starting vertex, it takes one second to jump to each successive vertex. If the mouse picks each vertex as the starting vertex with equal probability what is the expected number of seconds it will take to reach vertex 3? [3 pts]

How Many Seconds?	2.17

Put scratch work below. Scratch work is not graded but note that you should know how to explain your answer because you may be expected to do this on an exam.

$E_v = \left(\frac{1}{6}\right)(4 + 2 + 2 + 0 + 1 + 4)$

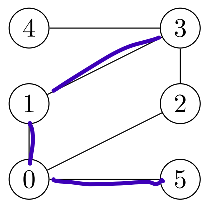
$6 + 2 + 5$

$8 + 5$

$E_v = \frac{1}{6}(13)$

$E_v = \frac{1}{6}(13) = \frac{13}{6} = \underline{2.17 \text{ seconds}}$

7. Given the following graph and cute little mouse:



Suppose the mouse starts at vertex x and does DFT according to the stack/linked-list implementation with pushing in decreasing numerical order.

(a) For each x give the order in which the vertices will be visited. [12 pts]

$x = 0$	0	1	3	2	4	5	2s
$x = 1$	1	0	2	3	4	5	3s
$x = 2$	2	0	1	3	4	5	3s
$x = 3$	3	1	0	2	5	4	0s
$x = 4$	4	3	1	0	2	5	1s
$x = 5$	5	0	1	3	2	4	3s

(b) Suppose that, starting at the starting vertex, it takes one second to jump to each successive vertex. If the mouse picks each vertex as the starting vertex with equal probability what is the expected number of seconds it will take to reach vertex 3? [3 pts]

How Many Seconds?	2

Put scratch work below. Scratch work is not graded but note that you should know how to explain your answer because you may be expected to do this on an exam.

Expected Value = Σ Value · Prob

each value being picked has a 1/6 chance so

$$E_v = \left(\tfrac{1}{6}\right)(2) + \left(\tfrac{1}{6}\right)(3) + \left(\tfrac{1}{6}\right)(3) + \left(\tfrac{1}{6}\right)(0) + \left(\tfrac{1}{6}\right)(1) + \left(\tfrac{1}{6}\right)(3)$$

$$= \frac{2}{6} + \frac{3}{6} + \frac{3}{6} + 0 + \frac{1}{6} + \frac{3}{6}$$

$$= \frac{5}{6} + \frac{3}{6} + \frac{1}{6} + \frac{3}{6} = \frac{6}{6} + \frac{6}{6} = \frac{12}{6} = 2 \text{ seconds}$$

8. Let q_i be the length of Q in the BFT algorithm after the ith iteration of the `while` loop. We'll say $q_0 = 1$ to be comprehensive since $Q = [s]$ when no iterations have completed. Of course there is some k such that $q_k = 0$ as this is when the algorithm ends.

   Suppose a graph has V vertices, where $V \geq 1$ is unknown.

   (a) What is the maximum value that q_i could be? Explain. [10 pts]

   Solution:

   Well the maximum value of qi would mean that the Queue is at its maximum value, which would only happen if you enqueued EVERY OTHER vertex in the graph, meaning that s would be adjacent to every vertex, so you would enqueue V-1 nodes.
   (minus one because you don't enqueue the current node, and in the BFT code you set x = Queue.dequeue(), so you dequeue before you enqueue all of the other elements)

   So the maximum value that qi could be is V - 1

   (b) What is the minimum nonzero value that q_i might be? Explain. [5 pts]

   Solution:

   The minimum nonzero value that qi can be is 1 because if you run this BFT algorithm on a single vertex, no other vertices will be enqueued, so the queue size will never change from the starting size of Q = [s] == 1.

   Or any case for that matter when you run the BFT algorithm on a vertex where there doesn't exist any vertices that are unvisited AND adjacent, so it won't enqueue anything.