Final Exam

Instructions:

- Do not start until directed to do so. Read all the instructions on this page.
- DO NOT write your name below. Write your Ashoka enrolment number. Then write that number at the top of **every** subsequent page.
- Write a random word of your choice on the first page. This will help us find you in case you write an incorrect enrolment number.
- **You are allowed two double-sided, A4-sized sheet with your own notes**. No calculators, programmable devices, or cell phones (or other communication devices) are permitted.
- Do not describe existing structures unless you want to modify them. E.g., you can just say "insert each value x into a min-heap" – there is no need to explain how a min-heap works.

Advice:

- You have 90 minutes to earn a maximum of 90 points (and 30 extra credit points). **Do not spend too much time on any single problem.** Read them all first, and attack them in the order that allows you to make the most progress. **It may be strategically best to not try to solve every problem.**
- When writing an algorithm, a **clear** description in English will suffice. Using pseudo-code is not required.
- Do not waste time re-deriving facts that we have studied. Simply state and cite them.
- Extra credit points will only apply to the exam portion of your grade.

Question	Points
True/False	15
Sort and Dijkstra	6
Search Trees	9
Graph Consulting	25
Cooking Online	35
EXTRA CREDIT: Suspicious Traveler	0
Total:	90

E.No.: _____ Random word: _____

Problem 1. [15 points] **True/False** (5 parts)

(a) [3 points] **T F** Inserting n elements into an empty dynamic array (via table doubling) takes worst case $O(n)$ time.

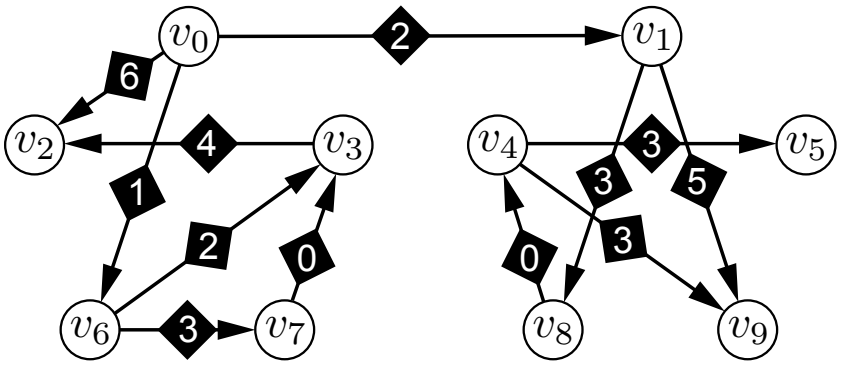
(b) [3 points] **T F** You are given a hash table implemented using a universal hash function where collisions are resolved via chaining. If you insert n elements into the empty hash table, the length of any chain can be at most $O(\log n)$.

(c) [3 points] **T F** Johnson's and Floyd-Warshall algorithms have the same asymptotic running time when applied to dense graphs.

(d) [3 points] **T F** If a graph contains no negative-weight cycles, there exists an ordering of edges such that Bellman-Ford computes shortest paths by relaxing each edge at most once.

(e) [3 points] **T F** Taking a sorted array and converting it into a min-heap takes $O(N)$ time; you should simply ignore the fact that the array has some a-priori order and run heapify.

Problem 2. [6 points] **Sort and Dijkstra** (3 parts)



Fill in the following tables based on the weighted directed graph above.

(a) [2 points] List the vertices in a topologically sorted order (many solutions exist).

v_0									

(b) [2 points] List the shortest path distance to each vertex from vertex v_0.

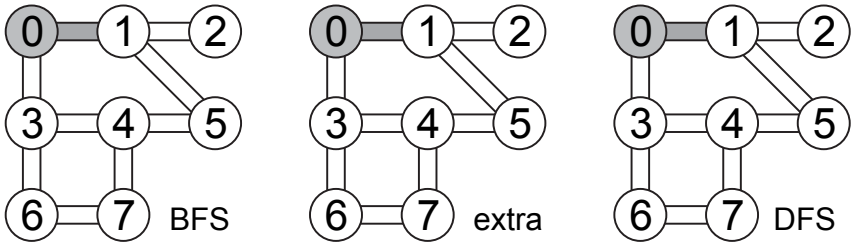
$\delta_0(v_0)$	$\delta_0(v_1)$	$\delta_0(v_2)$	$\delta_0(v_3)$	$\delta_0(v_4)$	$\delta_0(v_5)$	$\delta_0(v_6)$	$\delta_0(v_7)$	$\delta_0(v_8)$	$\delta_0(v_9)$
0									

(c) [2 points] List the vertices in the order in which Dijkstra would remove them from the priority queue in a search starting at v_0.

v_0									

Problem 3. [9 points] **Search Trees** (2 parts)

(a) [5 points] Below are three drawings of the same undirected graph. Color in edges corresponding to the tree of parent pointers constructed by a search of the graph, using breadth-first search on the left, and depth-first search on the right (there's a third in the middle in case you mess up). Begin your search starting at the filled in node **prioritizing vertices with lower indices** whenever ambiguity exists. The first edge has been colored in for you.



(b) [4 points] Describe a graph on n vertices for which the order of vertices first visited by both breadth-first and depth-first search could be the same.

Problem 4. [25 points] **Graph Consulting** (4 parts)
For each of the following four scenarios, choose a graph algorithm that best solves the problem. Specifically: describe a graph related to the problem, name an algorithm to apply to the graph, justify your choice of algorithm, and state the algorithm's running time. You do not need to explain algorithms; so you can just say "topological sort" or "Dijkstra" without explaining further.

(a) [6 points] A cell phone carrier transfers calls through a connected network of cell towers. Each tower can transfer calls to at most five other towers. Sending a call from one tower to another decreases the quality of the call by a known multiplicative factor between 0 and 1; different tower to tower transfers may decrease quality by a different factor. Help the carrier connect a call from one tower to another to maximize the quality of the connected call. Hint: What weights should you use? Consider the fact that logarithms convert products into sums.

(b) [7 points] Carah Sonnor is flying a drone in a park with her young son. After a while, the drone runs out of batteries and crash lands in a two foot tall hedge maze, whose walls are a subset of edges from a square grid. From the entrance of the maze, Carah can see where the drone landed and the layout of the maze. Help Carah direct her son along a fastest route through the maze to retrieve the drone.

(c) [6 points] San Holo is the debt-ridden pilot of a freight transport spaceship. He can fly directly between any two planets in the galaxy, though each one-way flight route demands its own fixed transport expenses. San can offset expenses by buying goods on one planet and selling them on another. His cargo hold can store at most one standard shipping container full of a single good, and local regulations require the sale of all cargo upon reaching any destination. Each planet buys or sells only a small set of possible goods at fixed prices. Find out if there is a way for San to **continually** make money in this trade network in order to pay off his debts.

(d) [6 points] A group of IIT-Delhi students decide to transfer to Ashoka. They want to drive to Ashoka along a shortest possible route measured by km driven, never looking back: that is, while driving, their straight-line distance to IIT-D should always **strictly increase**. Being IIT-D students, they are really bad at planning and math unless things are given to them in an MCQ format. Help them plan their trip, assuming there exists at least one route fitting their criteria.

Problem 5. [35 points] **Cooking Online** (2 parts)
Chulia Jild wants to make a responsive website where people can find and share recipes. Let R represent the number of recipes listed on the website at any given time. Each recipe has a name and contains a constant length description of ingredients, cooking instructions, and unit serving price for the meal.

(a) [15 points] Describe in detail a database that supports each of the following operations, describe how it supports each operation, argue that each operation achieves the stated running times. **State** whether **each running time** is a worst-case, expected, and/or amortized bound.

- `Add_Recipe`: add a recipe to the database in $O(\log R)$ time.
- `Find_Recipe`: return a recipe given its name in $O(1)$ time.
- `Similar_Price`: given a recipe name, return a list of 10 recipes closest in price to the given recipe in $O(\log R)$ time.

(b) [20 points] Each individual recipe contains at most 20 ingredients, though there may be many more ingredients contained in all recipes listed on the website. Chulia wants users to be able to search the website for ingredients.

Modify your database from part (a) so it supports **all three** operations from part (a), **in addition** to the following three. Describe how the new database supports each operation, and argue that each operation achieves the stated running times. **State** whether **each running time** is a worst-case, expected, and/or amortized bound.

- `Ingredient_Recipe`: return a list of recipes that use a given ingredient in $O(R')$ time, where R' is the number of recipes returned.
- `Best_Recipe`: given a list of N distinct ingredients, return the names of two recipes with highest and lowest price, that can be made using any subset of those ingredients in $O(R + N)$ time.
- `Saffron_Count`: return the number of recipes having saffron as an ingredient and price less than or equal to a queried price, in $O(\log R)$ time.

Problem 6. [0 points] **EXTRA CREDIT: Suspicious Traveler** (3 parts)
Srya Atark lives in Easteros, a nation of N towns connected by a network of T trails, where each trail connects two neighboring towns. Each town has a **unique numeric rank** representing its **loyalty** to the Queen, with the capital city the most loyal town of all. Srya has a map of Easteros, labeled with the length of each trail and the loyalty of each town. Townspeople are very suspicious. When Srya travels along a single trail from town A to town B, she will **attract suspicion** if town A is strictly less loyal to the queen than town B.

For each of the following three parts, describe a graph, and a graph algorithm applied to the graph, that will help Srya plan her route. Justify your choice of algorithm, and state your algorithm's running time. Each part is worth 10 extra credit points!

(a) [0 points] Srya Atark needs to travel from the capitol to her home in Summerfell to deliver a secret message to her sister. Describe an efficient algorithm to find the shortest length route from the capitol to Summerfell that **never** traverses a trail that would attract suspicion.

(b) [0 points] Having delivered her message to Summerfell, Srya must return to the capitol, which might be impossible to do without attracting suspicion. Thus she will take her time, spending one entire day sneaking along each trail between two cities. However, when she traverses a trail that attracts suspicion, she will spend three more days hiding in the city before moving on. Describe an efficient algorithm to find the minimum number of days that it will take Srya to return to the capitol.

(c) [0 points] Upon returning to the capitol, Srya steals a dragon egg from the Queen's dungeon, causing a panic. She must flee back to Summerfell as fast as she can, running along trails all the way back home at a fixed maximum speed S without rest. However, when entering a town from a trail that **attracts suspicion**, a person loyal to the Queen will fight her, delaying her travel by a fixed amount of time D. Describe an efficient algorithm to return Srya home as quickly as possible.