

CS101 Algorithms and Data Structures
Fall 2022
Homework 8

Due date: 23:59, November 20th, 2022

1. Please write your solutions in English.
2. Submit your solutions to gradescope.com.
3. Set your FULL name to your Chinese name and your STUDENT ID correctly in Account Settings.
4. If you want to submit a handwritten version, scan it clearly. **CamScanner** is recommended.
5. When submitting, match your solutions to the problems correctly.
6. No late submission will be accepted.
7. Violations to any of the above may result in zero points.

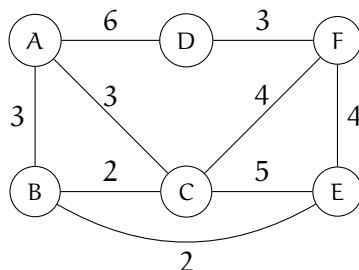
1. (12 points) Multiple Choices

Each question has **one or more** correct answer(s). Select all the correct answer(s). For each question, you will get 0 points if you select one or more wrong answers, but you will get 1 point if you select a non-empty subset of the correct answers.

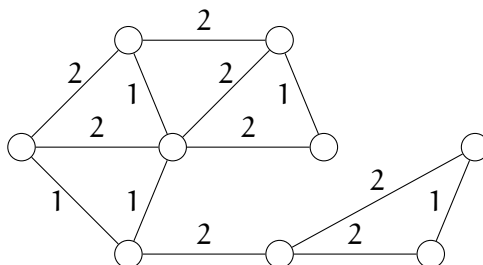
Write your answers in the following table.

(a)	(b)	(c)	(d)
ABD	BCD	C	AB

- (a) (3') Suppose we use the Prim's algorithm to find the minimum spanning tree of the following graph. Choose all possible sequences of edges added to the minimum spanning tree.



- A. {A, C}, {C, B}, {B, E}, {C, F}, {F, D}
- B. {A, B}, {B, C}, {B, E}, {C, F}, {F, D}
- C. {A, C}, {C, B}, {C, F}, {B, E}, {F, D}
- D. {A, B}, {B, E}, {B, C}, {E, F}, {F, D}
- (b) (3') Which of the following statements is/are true?
- A. The time complexity of the Prim's algorithm with a Fibonacci heap is always asymptotically better than that with a binary heap.
- B. The time complexity of the Prim's algorithm using adjacency list and binary heap is always better than that using adjacency matrix without a priority queue.
- C. The minimum spanning tree of a graph is unique if all the edges have distinct weights.
- D. The time complexity of the Kruskal's algorithm is $O(|E|\alpha(|V|))$ if we use the disjoint-sets with union-by-rank optimization and path-compression optimization.
- E. If T is a minimum spanning tree obtained by performing the Prim's algorithm starting with vertex v , then for any vertex u the path on the tree T connecting u and v is the shortest path from u to v in the graph.
- (c) (3') How many different minimum spanning trees does the following graph have?



- A. 4 B. 5 C. 6 D. 7

- (d) (3') Suppose $G = (V, E)$ is an undirected connected graph and that T is a minimum spanning tree of G . Define $w(e)$ to be the weight of e for $e \in E$. Which of the following statements is/are true?

A. If $C \subseteq E$ is a cycle in G and $e \in C$ is an edge on the cycle such that

$$\forall f \in C \setminus \{e\}, \quad w(e) > w(f),$$

then e does not belong to T .

B. Let $V = X \cup Y$ be a partition of V such that $X \cap Y = \emptyset$. Define

$$C(X, Y) = \{\{u, v\} \in E \mid u \in X, v \in Y\}.$$

If $e \in C(X, Y)$ is an edge such that

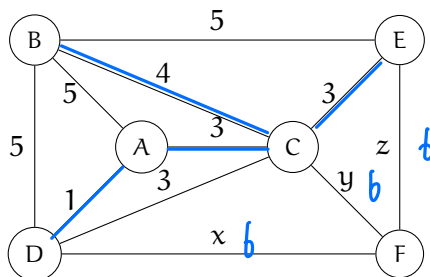
$$\forall f \in C(X, Y) \setminus \{e\}, \quad w(e) < w(f),$$

then e must belong to T .

- C. Suppose $T' \neq T$ is another minimum spanning tree of G . Let $w_0 \in \{w(e) \mid e \in T\}$ be the weight of some edge in T . Let m be the number of edges weighted w_0 in T . Then T' may contain less than m edges weighted w_0 .
- D. If $e \in E$ is an edge that has the largest weight among all edges in E , then e cannot belong to T .

2. (8 points) Minimum Spanning Tree

Consider the following weighted undirected graph.



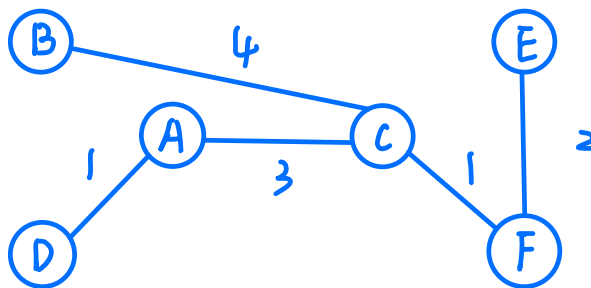
(a) (3') Suppose $(x, y, z) = (4, 1, 2)$ and that we use the Kruskal's algorithm to find a minimum spanning tree of the graph. To make your answer unique and clear, please follow the rules below.

- Use (u, v) to represent an undirected edge $\{u, v\}$, where $u < v$.
- Edges with same weight are sorted in alphabetical order. If two edges $e_1 = (u, v)$ and $e_2 = (w, t)$ have the same weight, e_1 appears before e_2 in the edge list if $(u < w) \vee ((u = w) \wedge (v < t))$.

Write down the sequence of edges added into the minimum spanning tree, and draw the tree.

Solution:

$\{A, D\}$
 $\{C, F\}$
 $\{E, F\}$
 $\{A, C\}$
 $\{B, C\}$



(b) (3') If $(x, z) = (2, 3)$, for what values of y is the edge $\{C, F\}$ guaranteed to be contained in a minimum spanning tree? Give a sufficient and necessary condition and briefly justify your answer.

Solution:

(1) $0 < y < 3$. the second step is to connect C and F
 (2) $y \geq 3$, if add the edge $\{C, F\}$ there will be a cycle $ADFC$. so $y \neq 3$
 So the condition is $0 < y < 3$

- (c) (2') If $5 \notin \{x, y, z\}$, is it possible for an edge weighted 5 to appear in a minimum spanning tree? Briefly justify your answer.

Solution:

NO. The edges of weight 5 are all related to B, but $\{x, y, z\}$ has nothing to do with B. When using Kruskal's algorithm, the possible cycles about B are ABC and CBE but the weight of $\{A, C\}$ $\{B, C\}$ $\{C, E\}$ are all smaller than 5, so the edge weighted 5 won't be chosen.

3. (9 points) Algebraic Geometry

Liu Big God, who loves pure math, has bought n books on algebraic geometry, the i -th of which has price a_i , $i = 1, \dots, n$. He will give his students some books to arouse their interest in pure math. For each student, Liu Big God is going to give him/her **one or two** books with total price not exceeding P .

Liu Big God is not going to keep any of these books, because he has read all of them. He wants to send all these books to students. What is the minimum number of students that can receive books?

It is guaranteed that $0 \leq a_i \leq P$ for every $i = 1, \dots, n$. You should come up with a greedy algorithm with time complexity $O(n \log n)$.

- (a) (3') Description of your algorithm in **pseudocode** or **natural language**.
- (b) (4') Proof of correctness of your algorithm.
- (c) (2') Time complexity.

Solution:

(a) Sort the books from largest to smallest. Then take the largest unassigned book and assign it to the kid with the least prices(The number of books doesn't exceed 2). Keep doing this until all the books are assigned. Each book assignment requires to sort the kids based on their total prices.

(b)Each time the book with the highest current price is given to the student with the lowest current price, this allocation increases the probability that one student will get two books, while reducing the number of people who will get the book.

(c)we have an iteration loop of size n (the number of books) and search for the kid with least prices takes $\log n$ time.

4. (9 points)

Given a set of $n \geq 3$ distinct positive numbers $S = \{s_1, s_2, \dots, s_n\}$, we want to find a permutation $A = \langle A_1, \dots, A_n \rangle$ of S , where $A_i \in S$ for all $i \in \{1, \dots, n\}$, such that

$$f(A) = A_1^2 + \sum_{i=2}^n (A_i - A_{i-1})^2$$

is maximized.

- (a) (3') Describe your algorithm that finds the permutation A for which $f(A)$ is maximized. Use **pseudocode** or **natural language**.
- (b) (4') Prove the correctness of your algorithm by showing that your choice on the value of A_1 is optimal, i.e. any other choice would not lead to a better solution.
- (c) (2') Time complexity. Your algorithm should be $O(n \log n)$.

Solution:

(a) Order n numbers in S from smallest to largest. A_1 chooses the median of S . Starting at A_2 , even digits are placed from the first digit of S to the largest, and odd digits from the last digit of S from the largest to the smallest. Do this until all the numbers have been placed.

(b) Choosing A_1 as the median in S after sorting increases the difference between even and odd digits in subsequent summations. This will make the sum of the square variances as large as possible.

(c) we have to sort set S with size n and search for the number that meets the demand to put in A_i which takes $\log n$ time.

5. (1 points) Discovery

- (a) (1') Let $G = (V, E)$ be an unweighted undirected graph where $V = \{v_1, \dots, v_n\}$ and $E = \{e_1, \dots, e_m\}$. For simplicity we assume there are no multiple edges (i.e. two or more edges incident to the same two vertices). Let $D \in \mathbb{R}^{n \times n}$ be the *degree matrix* whose (i, j) -th entry is

$$d_{ij} = \begin{cases} \deg(v_i), & \text{if } i = j, \\ 0, & \text{if } i \neq j. \end{cases}$$

Let $A \in \mathbb{R}^{n \times n}$ be the *adjacency matrix* of G , whose (i, j) -th entry is

$$a_{ij} = \begin{cases} 1, & \text{if } \{v_i, v_j\} \in E, \\ 0, & \text{if } \{v_i, v_j\} \notin E. \end{cases}$$

Note that $\deg(v_i) = \sum_{j=1}^n a_{ij}$. The matrix $L = D - A$ is the *Laplacian matrix*. Prove that L is positive semidefinite. (Hint: Try to show that $x^T L x \geq 0$ holds for every $x \in \mathbb{R}^n$.)

Solution:

$$\begin{aligned} x^T L x &= x^T (D - A) x = x^T D x - x^T A x \\ x^T L x &= \sum_{i=1}^n d_i x_i^2 - \sum_{i=1}^n \sum_{j=1}^n a_{ij} x_i x_j \\ &= \frac{1}{2} \left(\sum_{i=1}^n d_i x_i^2 - 2 \sum_{i=1}^n \sum_{j=1}^n a_{ij} x_i x_j + \sum_{j=1}^n d_j x_j^2 \right) \\ &= \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n (a_{ij} x_i^2 - 2a_{ij} x_i x_j + a_{ji} x_j^2) \\ &= \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n a_{ij} (x_i - x_j)^2 \geq 0 \end{aligned}$$

- (b) (0') STFW (Search The Friendly Web) about how the Laplacian matrix is related to the number of spanning trees of a graph.

Solution: