CS 270 Exam 1 True/False Repository

Neel Gupta

February 9, 2023

1. Given a binary max-heap with n elements, the time complexity of find the smallest element is $O(\log n)$.
2. A greedy algorithm considers the entire search space when making each step.
3. Suppose that for a graph $G = (V, E)$ the average edge weight is w. Then a MST of G will have weight at most $(V - 1)w$.
4. Consider two positively weighted graphs $G_1 = (V, E, w_1)$ and $G_2 = (V, E, w_2)$ with the same vertices and same edges s.t. every edge weighting in G_2 is the square of the edge weights in G_1. For any two vertices $u, v \in V$, any shortest path between u and v in G_2 is also a shortest path in G_1.
5. If all edges in a connected undirected graph have unit cost, then you can find the MST using BFS.
6. BFS is an example of a divide-and-conquer algorithm.
7. For any cycle in a graph, the cheapest edge in the cycle is in a MST.
8. DFS finds the longest paths from start vertex s to each vertex v in the graph.
9. The shortest path in a weighted DAG can be found in linear time.
10. Finding the k-th minimum element in an array of size n using a binary min-heap takes $O(k \log n)$ time.
11. We can merge any two arrays each of size n into a new sorted array in $O(n)$.
12. The shortest path in a weighted directed acyclic graph can be found in linear time.
13. Given a weighted planar graph, Prim's algorithm using a binary heap implementation will outperform Prim's algorithm using an array implementation.

14. If $f(n) = \Omega(n \log n)$ and $g(n) = O(n^2 \log n)$, then $f(n) = O(g(n))$.

15. Given a dense undirected weighted graph, the time for Prim's algorithm using a Fibonacci heap is $O(E)$.

16. In a binomial min-heap with n elements, the worst-case runtime complexity of finding the second smallest element is $O(1)$.

17. Let T be a complete binary tree with n nodes. Finding a path from the root of T to a given vertex $v \in T$ using BFS takes $O(\log n)$.

18. Dijkstra's algorithm works correctly on a directed acyclic graph even when there are negative-weight edges.

19. If the edge e is not part of any MST of G, then it must be the maximum weight edge on some some cycle in G.

20. If $f(n) = O(g(n))$ and $g(n) = O(f(n))$, then $f(n) = g(n)$.

21. The following array is a binary max-heap: [10,3,5,1,4,2].

22. There are at least 2 distinct solutions to the stable matching problem: one that is preferred by men and one that is preferrerd by women.

23. In a binary max-heap with n elements, the time complexity of finding the second largest element is $O(1)$.

24. Given a binary max-heap with n elements, the time complexity of find the smallest element is $O(\log n)$

25. Kruskal's algorithm can fail in the presence of negative cost edges.

26. If a weighted undirected graph has 2 MSTs, then its vertex set can be partioned into two, such that the minimum weight edge crossing the partition is not unique.

27. In a connected undirected graph, and using the same starting point, the depth of any DFS tree is at least as much as the depth of any BFS tree.

28. Algorithm A has runtime $\Theta(n^2)$ and algorithm B has runtime $\Theta(n \log n)$. From this we can conclude that A can never run faster than B on the same input set.

29. Let T be a complete binary tree with n nodes. Finding a path from the root of T to a given vertex $v \in T$ using BFS takes $O(\log n)$ time.

30. Amortized cost of operations in a Fibonacci heap is at least as good as the worst case cost of those same operations in a binomial heap.

31. Dijkstra's shortest path algorithm can be used to find shortest path in graphs with any edge weights.

32. Function $f(n) = 5n^2 4^n + 6n^4 3^n$ is $O(n^4 3^n)$.

33. Consider stable matching. Suppose Jack preferse Rose to others, and Rose prefers Jack to others. The pair (Jack, Rose) exists in every stable matching.

34. A DFS tree is a spanning tree.

35. A binary max-heap can be built using an unsorted list of elements in $O(n)$ time.

36. For some graphs BFS trees and DFS tress can be the same.

37. The number of cycles in a bipartite graph may be odd.

38. Stable matching algorithm presented in class is based on the greedy technique.

39. To delete the ith node in a binary min-heap, you can exchange the last node with the ith node, then check the nodes below the ith node to see if the ith node should move down the heap to "re-heapify" it.

40. If a connected undirected graph G has the same weights for every edge, then a minimum spanning tree can be found in linear time.

41. Given n numbers, one could construct a binary heap using the n numbers, then using the binary heap produce a sortest list of the numbers in $O(n)$ time.

42. In a Fibonacci heap, the insert operations has an amoritzed cost of $O(1)$ time, but the worst case cost is higher.

43. Function $10n^{10}2^n + 3^n \log n$ is $O(n^{10}2^n)$

44. A directed graph has a topological ordering if and only if it contains no cycle.

45. Both BFS and DFS can be used to find shortest path from one node to another node on graphs that are unweighted.

46. If a directed graph has a topological ordering, then this topological ordering is unique.

47. Kruskal's, Prim's, and Reverse-Delete algorithms are all examples of greedy algorithms.

48. In an unweighted strongly connected (directed) graph, the shortest distance from A to B is always the same as that from B to A.

49. In a weighted undirected graph, the shortest distance from A to B is always the same as that from B to A.

50. In an unweighted directed graph, the shortest distance from A to B is always the same as that from B to A.

51. In the stable matching problem with n men and n women, if a man and a woman are each other's last preferences, then they will never be matched with each other in any stable matching.

52. We can find the kth largest element in a binary max-heap in $\Omega(1)$ time.

53. A strongly connected (directed) graph cannot be a DAG.

54. If the heaviest weight edge e in an undirected connected graph G is unique, then e cannot belong to any MST of G.

55. The height of a complete binary tree with n nodes is $O(n)$.

56. Given a graph G, if there is no negative cost cycles in G, then Dijkstra's algorithm will work correctly on G.

57. In any graph, we have that $|E| = \Theta(|V|^2)$.

58. If path p is the shortest path from u to v and w is a node on the path, then the part of the path from u to w is also the shortest path from u to w.

59. Dijkstra's algorithm is able to find the shortest path in directed and undirected graphs with postitive edge weights.

60. Nodes in a binomial heap can have more than 2 children.

61. The following array is a binary max-heap: [16,14,10,10,12,9,3,2,4,1].

62. In the stable matching problem involving n men and n women, for any given set of preference lists, there will be at most two stable matchings.