Indian Institute of Technology Patna
Department of Computer Sc & Engg
November 2024
End Semester: CS5101/CS514/MC5101
Design and Analysis of Algorithms

Duration: 3 hours Full Marks = 100

Instructions:

- **Sections I and II should be answered in the main copy provided (not in the question paper).**
- **There is no negative marking**
- **No doubts will be entertained.**

Section-I

Context for questions 1-4

Algorithm 1 Enigma($A[0..n-1, 0..n-1]$)

Input: A matrix $A[0..n-1, 0..n-1]$ of real numbers;
for $i \leftarrow 0$ to $n-2$ **do**
  for $j \leftarrow I+1$ to $n-1$ **do**
    if $A[i,j] \neq A[j,i]$ **then**
      return false
    end if
  end for
end for
return true

1. What does this algorithm compute? (2 marks)
2. What is its basic operation? (1 mark)
3. How many times is the basic operation executed (best and worst-cases)? (2 marks)
4. What is the overall time complexity of this algorithm? (1 mark)
5. What happens when the backtracking algorithm reaches a complete solution? (1 mark)
   (a) It backtracks to the root
   (b) It continues searching for other possible solutions
   (c) It traverses from a different route
   (d) Recursively traverses through the same route
6. Which of the following algorithms can be used to most efficiently determine the presence of a cycle in a given graph ? (1 mark)
   (a) Depth First Search
   (b) Breadth First Search
   (c) Prim's Minimum Spanning Tree Algorithm
   (d) KruskalMinimum Spanning Tree Algorithm
7. The most efficient algorithm for finding the number of connected components in an undirected graph of n vertices and m edges has time complexity: (1 mark)
   (a) $\Theta(n)$
   (b) $\Theta(m)$
   (c) $\Theta(n+m)$
   (d) $\Theta(mn)$
8. Let S be an NP-complete problem, and Q and R be two other problems not known to be in NP. Q is polynomial-time reducible to S, and S is polynomial-time reducible to R. Which one of the following statements is true? (1 mark)
   (a) R is NP-complete
   (b) R is NP-hard
   (c) Q is NP-complete
   (d) Q is NP-hard
9. Choose the option with a function having the same complexity as a Fibonacci heap. (1 mark)
   (a) Insertion, Deletion
   (b) Minimum extraction, insertion
   (c) Insertion, Union
   (d) Union, delete
10. In a Fibonacci heap, when two heaps are merged, the root list of one heap is simply appended to the root list of the other heap. (True/False) (1 mark) T
11. Approximation algorithms are used for which type of problems? (1 mark)
   (a) NP-Hard problems
   (b) Sorting problems
   (c) Searching problems
   (d) P problems
12. In an approximation algorithm, the performance ratio is defined as: (1 mark)
   (a) The ratio of the approximate solution to the optimal solution
   (b) The number of iterations required.
   (c) The time complexity of the algorithm
   (d) The size of the input
13. Vertex-cover is a polynomial time 3-approximation algorithm. (True/False) (1 mark)
14. Consider approximation algorithm, Longest Processing Time rule is a 1.5 approximation algorithm. (1 mark)
15. Which data structure is used by Graham-scan algorithm in convex hull? (1 mark)
   (a) Queue
   (b) Hash Table
   (c) Heap
   (d) Stack

16. Identify the below algorithm/s that are not able to find the convex hull of a set of points? (1 mark)

(a) Graham's scan

(b) point-in-polygon algorithm

(c) Jarvis's march

(d) QuickHull.

17. Given two directed segments P_0P_1 and P_0P_2, then P_0P_1 is clockwise from P_0P_2 if the cross product is ...true....... (1 mark)

18. Which of the following are correct statements regarding splay trees? (2 marks)

A: Every single operation is guaranteed to be efficient.

B: Avoids worst-case linear time behavior of BST operations.

C: Guaranteed that a series of m operations will take O (m log n) time for a tree of $\cancel{m}$ n nodes.

(a) A, B, and C

(b) A and C

(c) B and C

(d) B only

19. Assume you are creating an array data structure that has a fixed size of n. You want to backup this array after every so many insertion operations. Unfortunately, the backup operation is quite expensive, it takes n time to do the backup. Insertions without a backup just take 1 time unit.

(a) How frequently can you do a backup and still guarantee that the amortized cost of insertion is O(1)? (1 mark)

(b) Prove that you can do backups in O(1) amortized time. Use the potential method for your proof. (3 marks)

20. The Breadth First Search algorithm has been implemented using the queue data structure. One possible order of visiting the nodes of the following graph in Figure 1 is: (3 marks)

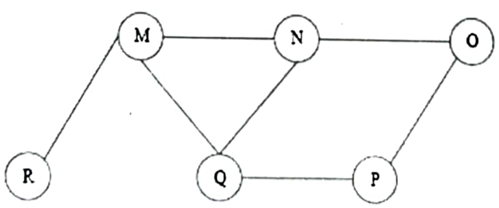


Figure 1: Graph

(a) MNOPQR

(b) NQMPOR

(c) QMNPRO

(d) QMNPOR

21. What are the amortized time complexities of the following Fibonacci heap operations: (3 × 1 mark = 3 marks)

(a) Extract Minimum

(b) Merge

(c) Decrease-Key

22. Using Ford-Fulkerson Algorithm the maximum possible flow in the graph in Figure 2 is: (4 marks)

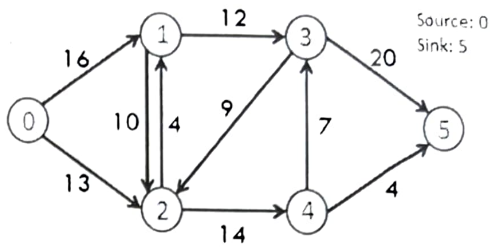


Figure 2: Graph

(a) 23

(b) 31

(c) 36

(d) 19

Context for questions 23-24

Lucky wants to study n flashcards for an exam using a deck of m flashcards (Assume that $n \leq m$). Lucky's algorithm for studying is as follows: Every morning he wakes up and randomly shuffles his flashcards. Then he goes over the flashcards one-by-one in the shuffled order. For every card he has studied before, he spends one hour reviewing it again. He stops when he sees a new flashcard, spends one hour studying it, and then he calls it a day. The next morning, he repeats this procedure until he learns a new flashcard, and so on until n days pass. For this problem, assume that random shuffling of cards takes negligible time. We want to study the runtime T which we define to be the total number of hours that Lucky spends studying.

23. Say $m = 4$ and $n = 3$. Assume that flashcards are numbered 1 through m. On the first day, Lucky goes over the flashcards in the (randomly chosen) order: [2,4,1,3]. He stops after the first flashcard 2, and calls it a day. On the second day, Lucky goes over the flashcards in the (randomly chosen) order: [3,1,2,4]. He stops after the first flashcard 3, and calls it a day. On the third day, Lucky goes over the flashcards in the (randomly chosen) order: [3,2,4,1]. He reviews flashcards 3 and 2 (he has seen them before), and then studies flashcard 4. He stops and calls it a day. In this example, T would be 1 + 1 + 3 = 5. In general, T depends on the random shuffles each day; in other words, T is a random variable. What is the worst case runtime of this algorithm? In other words, what is the worst case value of T for the worst possible sequence of shuffles? Express your answer in big-O notation in terms of n, AND briefly justify your answer. (5 marks)

24. Think of the expected runtime of this algorithm, that is $E[T]$, when the shuffles each day are uniformly and independently chosen random shuffles. If we keep n constant and increase the value of m, does this expected runtime increase, decrease, or stay the same? Justify. (4 marks)

25. Use branch and bound to solve the Travelling Sales Person problem in Figure 3. (*Note: Draw the final obtained branch and bound tree.*)(6 marks)

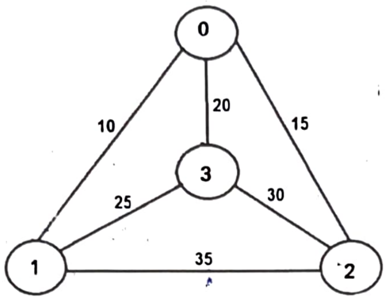


Figure 3: Travelling Sales Person

26. Perform a delete from the splay tree of Figure 4 for the key 9. Show each step. (8 marks)

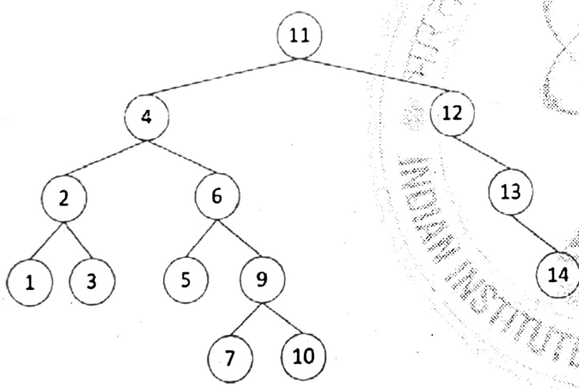
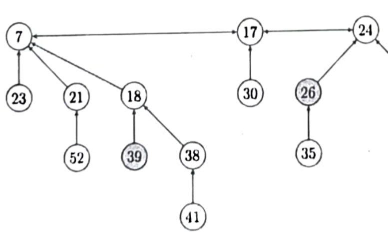


Figure 4: Splay Tree

27. Given the following Fibonacci heap in Figure 5: Perform



root ? & unmarked
root > be marked

Figure 5: Fibonacci heap

extract-min operation on the Fibonacci heap. Show all the steps. (6 marks)

28. Consider a stack that, in addition to push() and pop(), supports flush() which repeatedly pops items until the stack is empty.

(a) Using the potential function ϕ= number of items in the stack, show the amortized cost of each of these operations. (4 marks)

(b) What is the worst-case time complexity for an operation? (2 marks)

Section-II

1. What is principal n^{th} root of unity? Describe its properties. (2+3 marks)

2. Describe and prove Cancellation lemma. (4 marks)

3. Describe and prove Halving lemma. (6 marks)

4. Write the algorithm for determining whether two line segments intersect. (10 marks)

5. What is the lower bound on the running time of any Convex Hull algorithm? Prove this bound using the lower bound of comparison based sorting algorithms. (1+4 marks)