National University of Computer & Emerging Sciences, Karachi

Fall-2020 Department of Computer Science

Final Exam

19 December 2022, 01:00 PM – 04:00 PM

Course Code: CS2009	**Course Name:** Design and Analysis of Algorithm
Instructor Name / Names: Dr. Muhammad Atif Tahir, Dr. Farrukh Saleem, Dr. Waheed Ahmed, Miss Anuam Hamid, Miss Aqsa Zahid, Sohail Afzal	
Student Roll No:	**Section:**

Instructions:

- Return the question paper.
- Read each question completely before answering it. There are **11 questions** on **6 pages**.
- In case of any ambiguity, you may make assumption. But your assumption should not contradict any statement in the question paper.

Time: 180 minutes. **Max Marks: 50**

Question # 1 **[0.5*10 = 5 marks] [CLO 1]**

Answer the following questions. You must explain in only 3-4 lines.

(a) Write any two real life applications of the set cover problem
(b) If a problem does not have overlapping sub-problems, then dynamic programming approach works better than divide and conquer. True or False? Also briefly explain.
(c) Suppose we have five different points in one dimension. How can we find closest pair of points in *O (nlogn)*?
(d) Discuss any two real life applications of the closest pair problem.
(e) Suppose you want to lay pipelines in a building such that minimum pipe and other materials are consumed covering every desired location. Which technique from algorithms course will you employ and why?
(f) What is the advantage of Bellman ford algorithm over Dijkstra algorithm?
(g) What is negative weight cycle in a graph and how will you identify it through algorithm?
(h) What do you mean by small-o (o) and small-omega (w) in asymptotic notations?
(i) For a sparse graph, adjacency matrix is better than adjacency list. True or False. Briefly explain
(j) Can a problem be both NP and P ? Why or why not ?

Question # 2 **[0.25 * 12 = 3 marks] [CLO 2]**

Write time complexities of below mentioned algorithms.

Algorithm	Worst case time	Best case time
Insertion Sort		
Radix Sort		
Quick Sort		
Jump Search		
Brute force string matching		
Boyer Moore String Matching		

Question # 3 **[1*4 = 4 marks] [CLO 2]**

For each of the following questions, indicate whether it is T (True) or F (False), justifying by using some examples e.g. assuming a function

1. For all positive $g(n)$, $o\left(g(n)\right) \cap \omega\left(g(n)\right) = emptyset$
2. For all positive $g(n)$, If $g(n) + \omega\left(g(n)\right) = o\left(g(n)\right)$
3. f and g be two positive functions such that $f(n) = O\left(g(n)\right)$ then $f(n)^2 = O\left(g(n)^2\right)$
4. $2^{n+1} = \Theta\left(2^n\right)$

Question # 4 **[1 + 0.5 + 0.5 + 2 + 1 = 5 marks] [CLO 2]**

1 Define P, NP, NP-Hard and NP-Complete Problems.
2 Assuming P != NP, then why NP-complete ∩ P = ∅ is correct?
3 Which of the following is true about NP-Complete and NP-Hard problems.
   a. If we want to prove that a problem X is NP-Hard, we take a known NP-Hard problem Y and reduce Y to X
   b. The first problem that was proved as NP-complete was the circuit satisfiability problem.
   c. NP-complete is a subset of NP Hard
   d. All of the above
   e. None of the above

4 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. Then why R cannot be NP-Complete.

5 Identify which of the following belongs to P, NP, NP-Hard and NP-Complete

I. Integer Factorization
II. Vertex Cover
III. Graph Isomorphism
IV. 2-SAT

Question # 5 **[2 + 3 = 5 marks] [CLO 4]**

Consider the following Greedy Algorithm for set-cover problem Greedy Algorithm (SET-COVER):

(i) pick the set that covers the most points.
(ii) Throw out all the points covered.
(iii) Repeat (i) and (ii) until all points are covered

(a) Is it always possible to get optimal solution from above algorithm? Show with an example [2 marks]

(b) Proof that if the optimal solution uses k sets, the greedy algorithm finds a solution with at most *k* ln *n* sets. [2 marks]

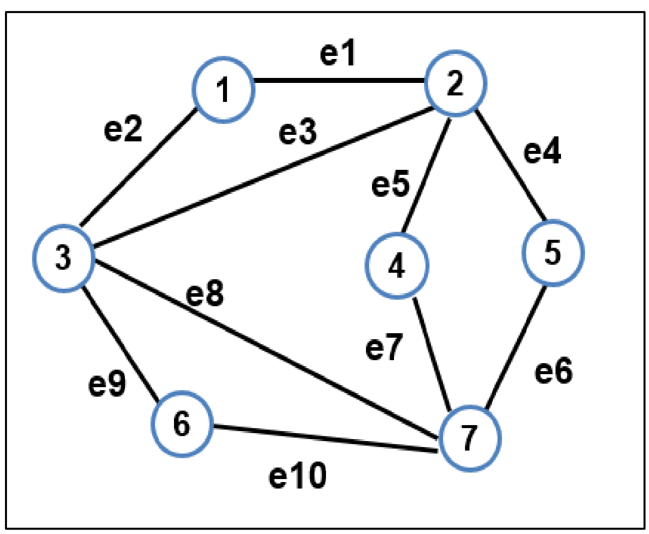
Question # 6 **[2.5 + 2.5 + 1 = 6 marks] [CLO 4]**

1. Prove that the 3-SAT problem reduces to the vertex cover problem, by constructing graph G of φ and show that φ is satisfiable iff G contains a vertex cover of size k and V-K = $|\varphi|$.

$$\varphi = (x_1 \lor \neg x_2 \lor \neg x_3) \land (\neg x_1 \lor x_2 \lor x_3) \land (x_1 \lor x_2 \lor x_3)$$

2. Prove that the vertex cover problem reduces to the set cover problem. Use the graph given below to show that the SET-COVER instance has set cover of size k iff G has a vertex cover of size k.

3. Prove that the **3-SAT** $\in$ ***NP***.



<u>Question # 7</u> **[2+2 = 4 marks] [CLO 4]**

Consider the following algorithm. Give answer of any two subparts (**from a to d)**.

Input : String S of "n" characters and String "P" of "m" characters, where m <= n
Output: index "i" where P begins as a substring of S

```
1:     for i = 0 to n-m
2:         hits = 0
3:         while (hits < m)
4:             if P[hits] != S[hits + i]
5:                 break
6:             hits = hits+1
7:         if (hits==m)
8:             print(i)
9:             break
```

a) Give an example of worst case input for this algorithm. Also tell exactly how many character comparisons are made (line 4) on your worst case input as a function of n and m?
b) For a fixed n, what value for m would maximize the cost of the worst case? Justify your answer through example input
c) Suppose that all characters in String "P" are different from each other. How would you accelerate the above algorithm to run in time *O (n)* on an "n" character string "S". Answer briefly in three to four lines.
d) If we remove line 9 in above code then what would be output of above code if S = ABAAABABAABABAA and P = BAA

<u>Question # 8</u> **[4 marks] [CLO 4]**

Points given below are sorted by polar angle. Show all steps to make convex hull on these points in *O (n)* time. (See appendix for formula of counter clockwise turn)

Sorted points [(0, 0), (7, 0), (3, 1), (5, 2), (9, 6), (1, 4)]

.

Question # 9 **[5 marks] [CLO 3]**

Apply the Bellman ford algorithm on the provided directed graph G to compute the shortest path from node "A" to all other nodes. Show all step to find the optimal cost and proof it optimal. In last, discuss the time complexity it take.

G =

	A	B	C	D	E	F
A	0	4	-	3	-	-
B	-	0	5	-	4	-
C	-	5	0	-	-	-2
D	4	-	-	0	3	-
E	-	-3	-	3	0	2
F	-	-	4	-	-	0

Question # 10 **[5 marks] [CLO 3]**

Professor Toole proposes a new divide-and-conquer algorithm for computing minimum spanning trees, which goes as follows.

- Given a graph G = (V, E), partition the set V of vertices into two sets V1 and V2 such that |V1| and |V2| differ by at most 1.
- Let E1 be the set of edges that are incident only on vertices in V1, and let E2 be the set of edges that are incident only on vertices in V2.

Solve a minimum-spanning-tree problem on each of the two subgraphs G1 = (V1, E1) and G2 = (V2, E2) recursively. Finally, select the minimum-weight edge in E that crosses the cut (V1, V2), and use this edge to unite the resulting two minimum spanning trees into a single spanning tree.

Either argue that the algorithm correctly computes a minimum spanning tree of G, or provide an example for which the algorithm fails.

Note: The minimum-weight edge that crosses the cut means, the minimum-weight edge that will combines V1 and V2.

Question # 11 **[2 + 2 = 4 marks] [CLO 2]**

Compute the time complexity for algorithms one by one. Show all steps.

void func(int n)
{
int i = 1;
while (i < n) {
int j = n;
while (j > 0) {
j = j / 2;
}
i = i * 2;
}
}
	void func(int n)
	{
	int count = 0;
	for (int i=0; i<n; i++)
	for (int j=i; j< i*i; j++)
	if (j%i == 0){
	for (int k=0; k<j; k++)
	printf("*");
	}
	}

Appendix

$$2 \times Area(a,b,c) = \begin{vmatrix} a_x & a_y & 1 \\ b_x & b_y & 1 \\ c_x & c_y & 1 \end{vmatrix} = (b_x - a_x)(c_y - a_y) - (b_y - a_y)(c_x - a_x)$$

- If area > 0 then a-b-c is counterclockwise.
- If area < 0, then a-b-c is clockwise.
- If area = 0, then a-b-c are collinear.

BEST OF LUCK