TOPIC 01: CRYPTOSYSTEM

RSA would be trivial to crack knowing the factorization into two primes of n in the public key, explain why RSA would be trivial to crack knowing $\phi(n)$.

How can we we decide whether a modular equation of the form $ax \equiv b \pmod{n}$ is solvable or not? If a solution exists, how can we calculate it?

What does EXTENDED-EUCLID (F_k, F_{k+1}) return? Prove your answer.

a) What is Euler Totient function? Mention it's applications.

b) Most computers can perform the operations of subtraction, testing the parity (odd or even) of a binary integer, and halving more quickly than computing remainders. Design an efficient binary gcd algorithm for input integers a and b, where $a \geq b$, that runs in O(log a) time. Assume that each subtraction, parity test, and halving takes unit time.

c) RSA would be trivial to crack knowing the factorization into two primes of n in the public key, explain why RSA would be trivial to crack knowing $\phi(n)$.

a) Write down the pseudocode of extended Euclid algorithm and analyze the computational complexity of your pseudocode.

b) Find all integers x that leaves remainders 2, 4, 5 when divided by 3, 5 and 7, respectively.

c) Consider an RSA key set with $p = 11$, $q = 29$, $n = 319$, and $e = 3$. What value of d should be used in the secret key? What is the encryption of the message $M = 100$.

Full Mark

What is modular inverse? What is the condition to have a modular inverse of an integer a mod m? Find $(197)^{-1}$ mod 3000.

Define $\phi(n)$. Calculate ϕ(n) where i) n is a prime, ii) n is a product of two co prime.

State Fermat's theorem and Euler's theorem. Show that Euler's theorem is generalized version of Fermat's theorem.

State the Chinese remainder theorem. Solve the following system of liner congruence using Chinese remainder theorem.

$$x \equiv 1(mod 5)$$
$$x \equiv 2(mod\ 6)$$
$$x \equiv 3(mod\ 7)$$

Write down the pseudocode of extended Euclid algorithm. Show that the number of recursive calls is $O(lgb)$.

The computational complexity of Euclidean algorithm for GCD is $O(\log_2 \min(a, b))$. Explain.

In an RSA cryptosystem, a particular A uses two prime numbers p = 13 and q =17 to generate her public and private keys. If the public key of A is 35. Then the private key of A is?

Solve $12x \equiv 17(mod\ 19)$.

Drive link [might get updated] : https://drive.google.com/file/d/1xhK3yMW9r7FlkI2aAHP6N_VqvD_gda6-/view?usp=sharing

TOPIC 02: BACKTRACKING

i) Write down a pseudocode to generate distinct permutation of a given string with duplicate characters with time complexity $O(n^2 * n!)$.

ii) Modify the above algorithm to generate all permutations of a given string without duplicate characters. Also show that the time complexity of the algorithm is $O(n * n!)$

Simulate the backtracking algorithm for the N queen problem where N=3.

b) **Demonstrate the benefits of using *Most Constrained Variable*, *Least Constraining Value* and *Forward Checking* heuristics in backtracking with a suitable example.**

What is backtracking approach? Someone give you a backtracking algorithm to generate all possible permutation for a given string. Is it possible to map the algorithm to solve graph coloring problem? Justify your position.

Drive Link [Might get updated] : https://drive.google.com/file/d/1Atf9s9dD5w9yueQF7SdpaYWkEbhLtuC1/view?usp=sharing

TOPIC 03: HASHING

X. a) What is direct addressing? Mention the time complexity to insert, delete and search of an item as well as space complexity in direct addressing. What are the limitations of direct addressing? [3]

b) Demonstrate what happens when we insert the keys 5, 28, 19, 15, 20, 33, 12, 17, 10 into a hash table with collisions resolved by chaining. Let the table have 9 slots, and let the hash function be $h(k) = k \bmod 9$. [2]

c) Draw the 11-entry hash table that results from using the hash function, $h(i) = (3i + 5) \bmod 11$, to hash the keys 12, 44, 13, 88, 23, 94, 11, 39, 20, 16, and 5, assuming collisions are handled by i) linear probing, ii) Quadratic probing and iii) double hashing (Assume that $h'(k) = 9 - (k \bmod 7)$). Explicitly state your necessary assumptions. [10]

3. What is clustering in hashing? How can we deal with this problem?

1. Compare and contrast between collision resolve by separate chaining and open addressing. 2

2. Which one of the following hash functions on integers will distribute keys most uniformly over 10 buckets numbered 0 to 9 for i ranging from 0 to 2020? 4

   i) $h(i) = i^2 mod\ 10$

   ii) $h(i) = i^3 mod\ 10$

   iii) $h(i) = (11 * i^2) mod\ 10$

   iv) $h(i) = (12 * i) mod\ 10$

3. A hash table of length 10 uses open addressing with hash function $h(k) = k\ mod\ 10$, and linear probing. After inserting 6 values into an empty hash table, the table is as shown below. 4

		42	23	34	52	46	33		
0	1	2	3	4	5	6	7	8	9

Determine the order in which the key values could have been inserted into the table.

2.	Consider inserting the keys 59, 88, 17, 28, 15, 4, 31, 22, 10 sequentially into a hash table of length m=11. Hashtable follows double hashing with h1(k) = (k+⌊m/2⌋)mod <your roll no> and h2(k) = 1 + (k mod (m-1)) You have to simulate the insertion procedure step by step.	[Marks:10]

2.	Suppose $\mathcal{H}$ forms a universal collection of hash functions. Calculate the expected insertion time of an arbitrary item with key k in the hash table with hash function randomly chosen from $\mathcal{H}$. Note that collision is resolved by chaining, and a new item is inserted at the end of the chain.	[Marks:10]

1 a) Let $\mathcal{H}_{pm} = \{h_{ab} : a \in \mathbb{Z}_p^*$ and $b \in \mathbb{Z}_p\}$ where [10]

- $m < p$, p is a prime number and $U \subset \mathbb{Z}_p = \{0, 1, 2, ..., p-1\}$
- $\mathbb{Z}_p^* = \{1, 2, ..., p-1\}$
- $h_{ab}(k) = ((ak+b) \bmod p) \bmod m$

Prove that $\mathcal{H}_{pm}$ is a universal class of hash functions.

b) Insert the keys 13, 9, 4, 51, 8, 43, 47 and 24 into a hash table of length m = 11 following open addressing with quadratic probing. [8]
Use $h'(k) = \lfloor (3k+1)/2 \rfloor$, $c_1 = 3$ and $c_2 = 7$.

c) How did the insertion task of question 1b) perform compared to the expected number of probes required for insertion of the same keys with uniform hashing? [5.3]

1) Suppose you have a 16-bit computing environment. Insert the following numbers in a hash table having 8 slots using the multiplication method. [10]

243, 1222, 23456, 11, 21

Note: You must show all calculations involved in finding the hash table slot for each number. Your calculation must not involve any floating-point arithmetic. Collision resolving will be done using the chaining method.

a) Insert the keys 10, 22, 31, 4, 15, 28, 17, 88 and 59 into a hash table of length 11 using double hashing with $h_1(k) = k$ and $h_2(k) = 1 + (k \bmod (m-1))$ [6]

b) "After completing insertions of question 1a searching for any key in the hash table should give a verdict (found at position x/not found) within 4 probes on an average"-Do you agree with this statement? Provide an explanation supporting your answer. [4]

c) Prove that in a hash table in which collisions are resolved by chaining, a successful search takes average-case time $\Theta(1+\alpha)$ under the assumption of simple uniform hashing. [4]

Drive Link [Might get updated] :
https://drive.google.com/file/d/1yPCYzd8e8ldtttLakn18nk8jI13hUqw7/view?usp=sharing

TOPIC 04: COMPUTATIONAL GEOMETRY

2. a) A disk consists of a circle plus its interior and is represented by its center point and radius. Two disks intersect if they have any point in common. Give an *O(nlgn)* time algorithm to determine whether any two disks in a set of n intersect. [5]

b) Consider the following function "orientation" which takes three ordered 2-D points as argument and returns 0 /1/2 if triplets are co-linear or makes a clock-wise or counter clock wise turn. [8]

```
int orientation(Point p, Point q, Point r)
{
   int val = (q.y - p.y) * (r.x - q.x) -
         (q.x - p.x) * (r.y - q.y);
   if (val == 0) return 0;  // colinear
   return (val > 0)? 1: 2; // clock or counterclock
}
```

You are given a set of 2-D points: S, your task is to simulate (all steps) Graham-Scan algorithm to find convex-hull from set S as given below. You can utilize "orientation" function in this purpose.

S ={(0, 3), (1, 1), (2, 2), (4, 4), (0, 0), (1, 2), (3, 1), (3, 3)}

c) Professor Jami proposes that only the x-dimension needs to be tested to determine ON-SEGMENT. Show why the professor is wrong. [2]

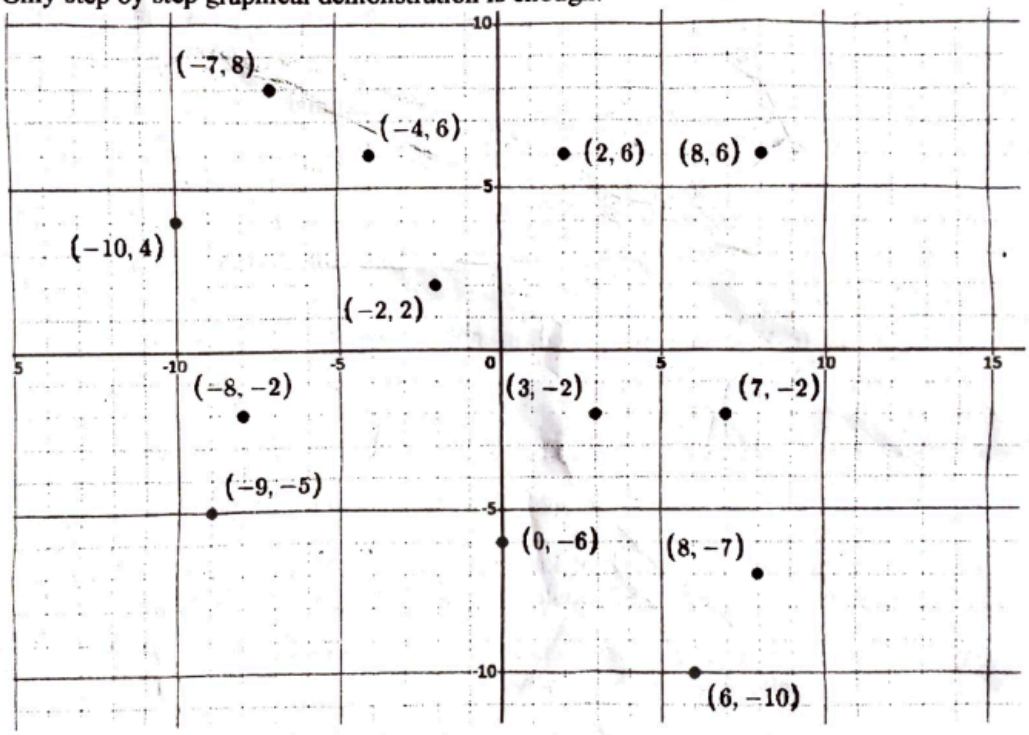
2 a) Prove that if $p_1 \times p_2$ is positive, then vector p_1 is clockwise from vector p_2 with respect to the origin $(0, 0)$ and that if this cross product is negative, then p_1 is counter clockwise from p_2. [5]

b) Professor Jami proposes that only the x-dimension needs to be tested to determine ON-SEGMENT. Show why the professor is wrong. [3.3]

c) Given a point $p_0 = (x_0, y_0)$, the right horizontal ray from p_0 is the set of points $\{p_i = (x_i, y_i): x_i \geq x_0 \text{ and } y_i = y_0\}$, that is, it is the set of points due right of p_0 along with p_0 itself. Show how to determine whether a given right horizontal ray from p_0 intersects a line segment p_1p_2 in O(1) time by reducing the problem to that of determining whether two line segments intersect. [7]

d) Write down the pseudocode of Graham Scan algorithm to generate a convex hull from a given set of points. Show the following two cases (with diagram)
i) include a point as a vertex of the convex hull and
ii) exclude a point from the vertex set of a convex hull. [8]

2 a) Consider the points plotted in the following figure. Find the convex-hull of these points using the Graham-Scan algorithm. You do not have to show calculations.
Only step by step graphical demonstration is enough. [6]



b) Prove that Kirkpatrick-Seidel's convex hull algorithm takes O(n lg h) time. [4]

c) Sketch a $O(n^2 \lg n)$ time algorithm to identify whether any three points in a set of n points are collinear. [4]

Drive Link [might get updated] :
https://drive.google.com/file/d/1Pm3_OwXNXheuqZUBpmKlwLK40ni0jh0K/view?usp=sharing

TOPIC 05: TRIE DATA STRUCTURE

Drive Link [might get updated] :
https://drive.google.com/file/d/1vpYy6lf6nPMruzjKWyjeX-bTKBbpXOf-/view?usp=sharing

TOPIC 06: STRING MATCHING

Drive Link [might get updated] :
https://drive.google.com/file/d/14mXqPeJA7x-H4hKmhyeS94cy8vIwH6Br/view?usp=sharing

TOPIC 07: RK SIR ASSIGNMENT

Drive Link [might get updated] :
https://drive.google.com/file/d/1VhJMnaS06Ea1vPJG1i7z5aRgoLCAo6ed/view?usp=sharing

TOPIC 08: APPROXIMATION ALGORITHMS

Drive Link [Might get updated] :
https://drive.google.com/file/d/1NNo2vbfBlYGVkuLDWt3yPZPQqPu86GR-/view?usp=sharing

TOPIC 09: NP COMPLETE PROOFS

https://docs.google.com/document/d/1FVEzzGRqf_n2q-dt5Q4X70K3YRMeMkGhCLu1c9AcSls/edit?usp=sharing