University of Dhaka

Department of Computer Science and Engineering

Incourse Examination

CSE 3203: Design and Analysis of Algorithm – II

Duration: 1 hr 20 minutes | Full Marks: 30

1. What is direct addressing? Mention the complexity to insert, delete and search of an item in direct addressing. Mention the space requirement for direct addressing. 3

2. Draw the 11-entry hash table that results from using the hash function, $h(i) = (2i + 5) \bmod 11$, 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 $h'(k) = 7 - (k \bmod 7)$). Explicitly state your necessary assumptions. 7

3. What is clustering in hashing? How can we deal with this problem? 2

4. Suppose that all characters in the pattern $P$ are different. Show how to accelerate naïve-string-matcher to run in time $O(n)$ on an $n$-character text $T$. 4

5. Give a linear-time algorithm to determine whether a text $T$ is a cyclic rotation of another string $T'$. For example, arc and car are cyclic rotations of each other. 4

6. Compute the prefix function for the pattern ababbabbabb. 3

7. 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. 5

8. What is branch and bound approach? To solve an optimization problem, you have two options- backtracking, and branch and bound. Choose an efficient one with proper justification. 2

Incourse Exam

CSE 3203: Design and Analysis of Algorithm – II

Duration: 1 hr 20 minutes

Full Marks: 30

1. What is modular inverse? What is the condition to have a modular inverse of an integer $a$ mod $m$? Find $(197)^{-1}$mod 3000. 3
2. Define $\phi(n)$. Calculate $\phi(n)$ where i) $n$ is a prime, ii) $n$ is a product of two co prime. 3
3. State Fermat's theorem and Euler's theorem. Show that Euler's theorem is generalized version of Fermat's theorem. 3
4. State the Chinese remainder theorem. Solve the following system of liner congruence using Chinese remainder theorem. 6

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

5. Write down the pseudocode of extended Euclid algorithm. Show that the number of recursive calls is $O(lgb)$. 5
6. Define class of NP problem. Prove that vertex cover problem is NP complete. 4
7. Define approximation algorithm and approximation ratio. Give an approximation algorithm of traveling sales man problem with triangle inequality. Prove that the algorithm is polynomial time 2-approximation algorithm. 6

**University of Dhaka**
**Department of Computer Science and Engineering**
**$3^{rd}$ Year $2^{nd}$ Semester B. Sc. Final Examination, 2018**
**CSE-3203: Design and Analysis of Algorithms - II**

**Total Marks: 60** **Time: 3 Hours**

**(Answer any four (4) of the following questions)**

1. 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]

2. a) i) Write down a pseudocode to generate distinct permutations of a given string with (possibly) duplicate characters with time complexity $O(n^2 * n!)$. [6]

   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 $O(n * n!)$. [4]

   b) Simulate the backtracking algorithm for $N$ queen problem where $N = 3$. [5]

3. a) Define cross product of two vectors. Consider two line segments $\overrightarrow{p_0p_1}$ and $\overrightarrow{p_0p_2}$ where common end vertex $p_0$ is not the origin; determine the cross product of these line segments. [3]

   b) i) Mention the conditions to intersect a pair of line segments. [3]

   ii) Professor J proposes that only the x-dimension needs to be tested to check whether a segment is on another segment. Show why the professor is wrong.

   c) In segment intersection algorithm, determine the condition when DIRECTION(x, y, z) procedure return i) greater than 0 ii) less than 0 iii) equal to 0. [3]

   d) Define convex hull. Write down the pseudocode for Graham Scan algorithm to generate a convex hull from a given set of points. Show that the pair of points farthest from each other must be vertices of Convex Hull(Q) where Q is the set of given points. [6]

4. a) Write down the pseudocode of extended Euclid algorithm and analyze the computational complexity of your pseudocode. [7]

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

   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 $M = 100$. [4]

5. a) Write down a pseudocode to determine a longest prefix of a string which is also suffix of the string. Using your algorithm, determine the longest prefix of the string "cgtacgttcgtacg" that is also a suffix of this string. [4]

   b) Construct the string-matching automaton for the pattern P = "aabab" and illustrate its operation on the text string T = "aaababaabaababaab". [4]

   c) We call a pattern P non-overlappable if $P_k \sqsupset P_q$ implies $k = 0$ or $k = q$. Describe the state-transition diagram of the string-matching automaton for a non-overlappable pattern. [2]

   d) Explain how to determine the occurrences of pattern $P$ in the text $T$ by examining the $\pi$ function for the string PT (the string of length m + n that is the concatenation of $P$ and $T$.) [5]

6. a) Define P, NP and NP-Complete with examples. [6]

   b) How can we prove a particular problem is NP-Complete? Describe the procedure in details with example. [9]