CS 101 *Data Structure and Algorithm*

Fall 2019

Final Exam Sample

INSTRUCTOR: Dengji Zhao, Yuyao Zhang

INSTRUCTIONS

- You have 2 hours.
- You are not allowed to bring any papers, books or electronic devices including regular calculators.
- You are not allowed to discuss or share anything with others during the exam.

Name	
ID	
Room	
Seat Number	
All the work on this exam is my own. **(please sign)**	

THIS PAGE INTENTIONALLY LEFT BLANK

1. **(20 points) Multiple Choice (Choose all correct answers)**

You should write your answer in the box below

Question(a)	Question(b)	Question(c)	Question(d)

(a) **(5 pt)** You are given a hash table with n keys and m slots. Which of the following statement(s) is(are) correct?

(A) The advantages of chained hash table over the open addressing hash table are that deletion is easier and space used is less.

(B) If collisions are resolved by linear probing, with simple uniform hashing, and assume that the table is initially empty. The probability that the first two slots of the table are filled after the first two insertions is $\frac{3}{m^2}$.

(C) If collisions are resolved by linear probing, and we maintain $m = \Theta(n)$, then we can always expect a good search and insert runtime.

(D) If collisions are resolved by quadratic probing, then it satisfies the ***uniform hashing assumption***: the probe sequence of each key k is equally likely to be any of the $m!$ permutations of the slots $\{0, 1, \cdots, m-1\}$.

(E) None of the above statements is correct.

(b) **(5 pt)** There are 1000 elements remaining to be sorted. With the requirement that the sorting should be fast and stable, we choose ______ to be our sorting algorithm. In another case, we choose ______ to be our sorting algorithm with the requirement that the space it takes is as little as possible.

(A) Insertion Sort

(B) Merge Sort

(C) Quick sort

(D) Heap Sort

(c) **(5 pt)** Which statement(s) is(are) **not** true about shortest path algorithms?

(A) Dijkstra's algorithm is based on Greedy paradigm and Floyd-Warshall Algorithm is based on Dynamic Programming paradigm.

(B) Dijkstra's algorithm is based on Dynamic Programming paradigm and Floyd-Warshall Algorithm is based on Greedy paradigm.

(C) Dijkstra's algorithm may not terminate if the graph contains negative-weight edges.

(D) Floyd-Warshall algorithm can be used to solve the APSP problem in all graphs correctly.

(d) **(5 pt)** Which statement(s) is(are) true about NP and reuduction?

(A) The problem of determining whether there exists a Hamiltonians cycle in an undirected graph is in NP.

(B) The abbreviation NP stands for Not Polynomial time.

(C) $P \subseteq NP$.

(D) None of the above.

2. (6 points) Recursive Runtime Analysis

(a) (2 pt) $T(n) = 8T(n/2) + O(n^2)$

(b) (2 pt) $T(n) = 4T(n-2)$

(c) (2 pt) $T(n) = T(n - n^{1/3}) + 1$

3. (10 points) Animal Protection

We have n endangered animals but unfortunately, k of them has infected a disease. We need to identify these k animal so as to treat them.

The infection can be detected by testing the animal's blood. The test takes long time so that we need to minimize the number of test carried out.

We can mix the blood samples from a subset of animals. If at least one of the animals in the subset is infected, then the test result is Yes. Otherwise, the result is NO.

Design an algorithm that uses $O(k \log n)$ tests to find all infected animals.

4. (16 points) Discount of the train

Steven and Olivia attend different colleges in different cities so that in summer holiday, Steven wants to take train to go from his city s to Olivia's city t.

The list of train can be describe as a directed graph $G = (V, E)$ and $\{w_e | e \in E\}$ represents the cost of every single travel of train. Steven has two coupons, each of which can be used for a 50% discount on any single train.

Define a shortest path problem whose solution can be used to find the route from s to t with minmal spending. Describe the set of vertices and edges in your problem. Prove the correctness and state its running time in terms of $|V|$ and $|E|$.

5. (16 points) Huffman Coding

In this question we will consider how much Huffman coding can compress a file F of m characters taken from an alphabet of $n = 2^k$ characters $x_0, x_1, \ldots, x_{n-1}$ (each character appears at least once).

(a) (4 pt) Let $S(F)$ represent the number of bits it takes to store F without using Huffman coding (i.e., using the same number of bits for each character). Represent $S(F)$ in terms of m and n.

(b) (5 pt) Let $H(F)$ represent the number of bits used in the optimal Huffman coding of F. We define the *efficiency* $E(F)$ of a Huffman coding on F as $E(F) := S(F)/H(F)$. For each m and n **describe a file** F for which $E(F)$ is as small as possible.

(c) (7 pt) For each m and n describe a file F for which $E(F)$ is as large as possible. How does the largest possible efficiency increase as a function of n? Give your answer in big-O notation.

6. (16 points) Dynamic Programming

(a) There are n types of toys that you with to collect. Each time you buy a toy, its type is randomly determined from a uniform distribution (i.e., all possible types have equal probabilities). Let $p_{i,j}$ be the probability that just after you have bought your ith toy, you have exactly j toy types in your collection, for $i \geq 1$ and $0 \leq j \leq n$.

i. (3') Find a recursive equation of $p_{i,j}$ in terms of $p_{i-1,j}$ and $p_{i-1,j-1}$ for $i \geq 2$ and $1 \leq j \leq n$.

ii. (3') Describe how the recursion from (a) can be used to calculate $p_{i,j}$.

(b) (10') Given n objects and a knapsack, item i weighs $w_i > 0$ kilograms and has value v_i where $n > v_i > 0$. The knapsack has capacity of W kilograms. The numbers n, v_i are integers and w_i, W are real numbers. What is the maximum total value of items that we can fill the knapsack with? Design an efficient algorithm. For comparison, our algorithm runs in $O(n^3)$.

7. (16 points) Independent Set

Prove that the problem of **Independent Set** is NP-complete. The definition of the problem is stated as following. (Hint: 3-SAT is NP-complete.)

Independent Set (IS): Given an undirected graph $G = (V, E)$ and an integer k does G contain a subset V' of k vertices such that no two vertices in V' are adjacent to one another.