University of Waterloo
CS240 Winter 2024
Assignment 4

Due Date: Tuesday, March 19 at 5:00pm

Please read `https://student.cs.uwaterloo.ca/~cs240/w24/assignments.phtml#guidelines` for guidelines on submission. **Each question must be submitted individually to MarkUs as a PDF** with the corresponding file names: a4q1.pdf, a4q2.pdf, ... , a4q7.pdf . It is a good idea to submit questions as you go so you aren't trying to create several PDF files at the last minute.

Late Policy: Assignments are due at **5:00pm**, with the grace period until 11:59pm.

Question 1 [5 marks]

Suppose we have an array S of n numbers and use only the equality ('=') to compare whether 2 numbers are equal or not. No other comparisons (such as '$<$' or '$>$' or '$\leq$' or '$\geq$') are allowed. Given S, we are interested in the `EqualPair(S)` problem. If S has at least a pair of equal elements then the answer to `EqualPair(S)` is the indexes of equal elements, otherwise the answer is -1. For example, for `EqualPair([1,1,2,3,3])`, one possible answer is $(0, 1)$ since the first two elements of array S are equal, and for for `EqualPair([1,2,4,6,3])` the answer is -1. Derived the exact (not asymptotic) bound for the number of equality comparisons required to solve the problem on sequence S of size n. Your bound should be as tight as possible.

Question 2 [(2+2)+2+3=9 marks]

a) Let $S(n) = \{(0011)^k\$ | k = 1, 2, ...n\}$, where w^k operation repeats w exactly k times. For example, $S(2) = \{0011\$, 00110011\$\}$.

- **i)** What is the height of the compressed trie storing $S(n)$? Prove your claim by induction.
- **ii)** Recall that each internal node of the compressed trie stores an integer index. Derive the recurrence relationship for the sum of indexes over all internal nodes in a compressed trie storing $S(n)$. Solve the recurrence relationship and give the exact (not asymptotic) solution.

b) Let T be a standard trie (binary alphabet, no prunning or compression). Suppose the longest word stored in T has length m. What is the maximum possible number of nodes in T?

c) Explain how to modify the standard trie (including operations of insertion and deletion, if needed) so that we can find the number of odd-length words in the trie which have prefix w in $O(|w|)$ time.

Question 3 [1+1+2+2=6 marks]

Consider a hash dictionary with table of size $M = 10$. Suppose items with keys $k = \{4371, 1323, 6173, 4199, 4344, 1679, 1989\}$ are inserted in that order using hash function $h_1(k) = k(\text{mod } 10)$. Draw the resulting hash table if we resolve collisions using

a) Chaining

b) Linear probing

c) Double hashing with the secondary hash function $h_1(x) = \lfloor \frac{x}{1000} \rfloor$.

d) Cuckoo hashing with the second hash function $h_1(x) = \lfloor \frac{x}{1000} \rfloor$.

Question 4 [1+2+2+3=8 marks]

Solve this problem under Uniform Hashing Assumption and using hashing with chaining, where hash table has size M.

a) What is the probability that two keys k_1, k_2 hash into location 0 of the hash table?

b) What is the probability that two keys k_1, k_2 hash into the same location the hash table?

c) What is the probability that n keys $k_1, k_2, ..., k_n$ hash into the same location the hash table?

d) What is the probability that key k_1 hashes into the same location as at least one of the keys $k_2, ...k_n$?

Question 5 [6 marks]

Design an algorithm that given an array A of size n storing integers, and a number m returns indices $i \leq j$ such that $\sum_{k=i}^{j} A[k] = m$. If no such indices i, j exist, your algorithm should return null. If there is more than one pair of i, j satisfying the condition, your algorithm can return any such pair. For example, $A = [5, 7, 9, 11, 13, 15]$ and $m = 33$, your algorithm should return $i = 2, j = 4$. Your algorithm must have $O(n)$ expected running time. You can explain your algorithm in English or write pseudocode.

Question 6 [(2+2)+4= 8 marks]

a) Suppose quadtree stores the following points:

$$(127, 15)\ (6, 127)\ (1, 1)\ (80, 100)\ (40, 30)\ (35, 42)\ (35, 49)\ (2, 3).$$

i) What is the side length of the smallest square (subdivision)?

ii) What is the height of the quadtree?

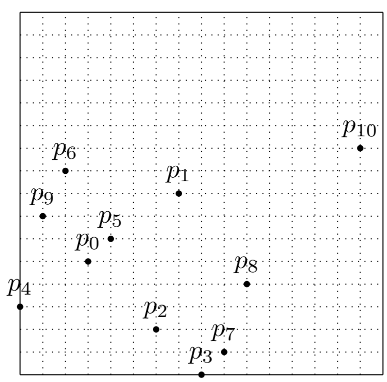
b) What is the largest number of points a quadtree with $4m + 1$ total number of nodes can store?

Question 7 [4+4+ (2+2+2) =14 marks]

Consider the following set of points in two dimensions:

$$S = \{(3, 5), (7, 8), (6, 2), (8, 0), (0, 3), (4, 6), (2, 9), (9, 1), (10, 4), (1, 7), (15, 10)\}.$$

a) Draw the plane partition diagram and kd-tree.



b) Show how a search for the points in $R = [1.6 \leq x < 6.5, 4.5 \leq y < 11.5]$ would proceed. More specifically, describe the 'color' (blue, green, red) of nodes as done in the handouts for module 8 (slide 21). You can directly color the nodes in the picture you drew in (a).

c) The goal of this exercise is to show that splitting kd-tree both vertically and horizontally is crucial for efficient search. Consider a modified kd-tree which we will call a degenerate kd-tree. A degenerate kd-tree uses only vertical splits, never horizontal. For example, let $S = \{(1, 2), (2, 3), (3, 4), (4, 5)\}$. Then a degenerate kd-tree first splits the space into two regions with line $x = 3$, with points $(1, 2), (2, 3)$ on the left and $(3, 4), (4, 5)$ on the right. Then the left region is split with a line $x = 2$, and the right region with the line $x = 4$. Let $S = \{(i, i)\ |i \in \{1, ..., n\}\}$. Let T be a degenerate tree on S. You can assume n is a power of 2.

i) What is the height of T? Explain your answer.

ii) How many nodes does T have? Explain your answer.

iii) Give an example of a range search on S such that it visits all the nodes of the degenerate kd-tree yet the search result is empty. Explain your answer.