CSCI2100: Quiz 2

Name: Student ID:

Note: A multiple-choice question has only one correct answer unless otherwise stated.

Problem 1 (10%). After applying the following operations to an empty stack:

push(35), push(36), push(43), push(8), pop, pop, push(51), pop

what is the content of the stack? Answer: [ ]

A. 35, 36
B. 36, 43
C. 35, 8
D. 35, 51

Answer: *A*

Problem 2 (10%). After applying the following operations to an empty queue:

enqueue(35), enqueue(36), enqueue(43), enqueue(8), dequeue, dequeue, enqueue(51), dequeue

what is the content of the queue? Answer: [ ]

A. 8, 51
B. 36, 43
C. 35, 8
D. 35, 51

Answer: *A*

Problem 3 (10%). Identify the operations below that can be performed in $O(1)$ cost (including $O(1)$ expected cost). There is more than one correct choice; no marks are given unless you can identify all of them. Answer: [ ]

A. Push an element into a stack.
B. Insert an element into a linked list.
C. Dequeue an element from a queue.
D. Determine whether the value 10 is in a hash table.

Answer: $ABCD$

Problem 4 (10%). Which of the following are true? There is more than one correct choice; no marks are given unless you can identify all of them. Answer: [ ]

A. Consider a data structure that supports a certain operation in $O(1)$ amortized time. Then, any sequence of n such operations requires $O(n)$ worst case time, regardless of the value of n.
B. Consider a data structure that supports a certain operation in $O(1)$ amortized time. But still, it is possible for the structure to take $O(n)$ time to process *one* operation, where n is the number of operations that have already been processed.
C. There is a hash function that can guarantee $O(1)$ expected dictionary search on all input sets.
D. In a tree, the number of internal nodes cannot exceed that of leaf nodes.

Answer: AB

Problem 5 (10%). Consider $S = \{1, 3, 4, 5, 7, 8, 9, 10, 11, 12, 13, 15\}$. We build a hash table on S with hash function $h(k) = 1 + k \bmod 4$. If we look up an element q in the hash table, which of the following value of q has the lowest look up cost? Answer: [ ]
A. 24 B. 23 C. 22 D. 21

Answer: C

Problem 6 (20%). Consider the hash function $h(k) = 1 + k \bmod 4$. Give a set S of 10 integers to meet both conditions below:

- If we build a hash table on S using $h(k)$, then all the integers of S fall in the same bucket (recall that a *bucket* contains all the elements of S having the same hash value).
- The aforementioned bucket is the one we probe in order to look for integer 35.

Answer: $S = \{3, 7, 11, 15, 19, 23, 27, 31, 35, 39\}$.

Problem 7 (30%). Let S be a set of $n \geq 2$ distinct integers where n is a power of 2. The set S is given in an array that has not been sorted. Give an algorithm to find the $\log_2 n$ largest integers of S in $O(n \log \log n)$ time. For example, if $S = \{30, 50, 10, 90, 80, 20, 70, 60\}$, then you should output 70, 80, 90.

Note: by using k-selection, we can solve the problem in $O(n)$ expected time. However, here our $O(n \log \log n)$ time bound needs to hold deterministically. If you want to use a deterministic k-selection algorithm, you must describe the algorithm in full (because it has not been covered in this course). This problem admits an elegant solution that does not require k-selection.

Answer 1: Initialize an empty priority queue H (min-heap). Process each element $e \in S$ in turn as follows. First, insert e to H. Then, check if H has more than $\log_2 n$ elements; if so, perform a delete-min (after which H will have exactly $\log_2 n$ elements). After all the elements of S have been processed, report the elements of H in ascending order with $\log_2 n$ delete-mins.

Answer 2: Construct a max-heap from S in $O(n)$ time. Then, perform $\log_2 n$ delete-max operations and returned the elements found. The total cost is $O(n + \log^2 n) = O(n)$.