Student Name: ______________________ ________ J#: ___________________

CSC 323 Algorithm Design and Analysis, Spring 2019
Instructor: Dr. Natarajan Meghanathan

Exam 1 (Take Home) **Total: 100 points**

Due: March 7th, 2019 (1 PM, in-class). Exam solutions submitted after 1 PM will not be accepted. Print this exam, answer in the space provided (use additional sheets, if needed), staple everything together and submit.

Note: Strictly, there should NOT be any copying. If the instructor finds that two or more exam solutions involve some sort of copying, all the concerned students found to be involved in copying will get a zero.

Q1 - 12 pts) We will define a unimodal array as an array of 'distinct' integers wherein the array is a sequence of monotonically decreasing integers followed by a sequence of monotonically increasing integers. Design a $\Theta(\log n)$ algorithm to determine the **minimum element** in the unimodal array.

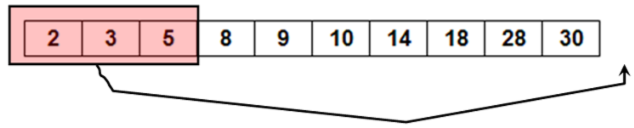
(a) Show the pseudo code of your algorithm.
(b) Justify the correctness of the algorithm.
(c) Analyze the run-time complexity of the algorithm and show that it is $\Theta(\log n)$.
(d) Show the working of the algorithm (along with the appropriate index values) for three different cases (in each case, the array should be of size at least 10 integers):

i) the array is a sequence of monotonically decreasing integers followed by a sequence of monotonically increasing integers
ii) the array is strictly a sequence of monotonically decreasing integers
iii) the array is strictly a sequence of monotonically increasing integers

Student Name: ________________________ _________ J#: ____________________

Q2 - 20 pts) Consider a sorted, but rotated array of distinct integers (i.e., no two integers are the same). For example, the following sorted array

2	3	5	8	9	10	14	18	28	30



when rotated three elements to the "left" becomes:

0	1	2	3	4	5	6	7	8	9
8	9	10	14	18	28	30	2	3	5

All the elements (except an element called the pivot at index p) of the sorted, but rotated array of integers have a property that they are less than the element to the right of them. Only the pivot element is greater than the element to the right of it. In the above sorted, but rotated array of integers, the pivot is the integer 30 at index 6. Incidentally, the pivot also happens to be the largest element in the array and is the last element in the original sorted array (before the rotation). In the above example, the pivot element 30 is the largest element of the array and is also the last element of the original sorted array (before the rotation).

(a) Design a binary search-based algorithm to identify the pivot in a sorted, but rotated array of integers.
(b) Extend the algorithm of (a) to do a successful search for a key that is present in the sorted, but rotated array.
(c) Extend the algorithm of (a) to do a unsuccessful search for a key that is not present in the sorted, but rotated array.
(d) For each of the algorithms in (a), (b) and (c), illustrate the execution of the algorithm for the array given in the problem statement.
(e) Analyze the time complexity of the algorithms of (a), (b) and (c).

Note that in addition to describing the working of your algorithms, you should also write the pseudo code for your algorithms of (a), (b) and (c).

Student Name: ______________________ ________ J#: ____________________

Q3 - 15 pts) The number of inversions is an array is the number of (i, j) pairs (where i and j are index positions and each pair is considered only once) such that A[i] > A[j] and i < j.

For example, the following array has 5 inversions as shown.

Index	0	1	2	3	4
Array, A	**10**	**50**	**40**	**20**	**30**

Index Pairs	Inversions
(1, 2)	A[1] > A[2]
(1, 3)	A[1] > A[3]
(2, 3)	A[2] > A[3]
(1, 4)	A[1] > A[4]
(2, 4)	A[2] > A[4]

(a) Modify the pseudo code of the Insertion Sort algorithm so that it can compute the number of inversions in an array. Write the modified pseudo code and justify your modification.
(b) Analyze whether the modification would have any impact on the asymptotic time complexity of Insertion sort.
(c) Run your modified version of the Insertion sort algorithm on the above array and show that it can determine the number of inversions to be 5.

Q4 - 14 pts) Develop a recursive version of the Bubble Sort algorithm.
(a) Write the pseudo code of the algorithm and justify that it is recursive and works correctly.
(b) Write the recurrence relation for the algorithm and solve it using one of the two approaches discussed in class, as appropriate. Solve the recurrence relation and show that the time complexity of the recursive algorithm is $\Theta(n^2)$.

Student Name: ______________________ ________ J#: ___________________

Q5 - 14 pts) Let the hash function be H(K) = K mod 5. Given two sets A and B, the Jaccard Index of A and B, J(A, B) is defined as follows.

$$J(A,B) = \frac{|A \cap B|}{|A \cup B|}$$

(a) Design a hash table based algorithm to determine the intersection of two sets A and B. Write the pseudo code.
(b) Show the execution of the algorithm in (a) on the sets A and B assigned to you and determine the intersection of the two sets. Determine the total number of comparisons encountered.

(c) Design a hash table based algorithm to determine the union of two sets A and B. Write the pseudo code.
(d) Show the execution of the algorithm in (c) on the sets A and B assigned to you and determine the union of the two sets. Determine the total number of comparisons encountered.

(e) Use the formula shown above to determine the Jaccard Index of the two sets A and B assigned to you.

Student Name	**Set A**	**Set B**
Demetrius Brown	[20, 21, 18, 13, 14, 19]	[11, 20, 15, 21, 19, 18]
Jahelle Cato	[17, 16, 18, 13, 15, 14]	[13, 15, 20, 21, 10, 14]
Nzefili Chukwuma	[19, 21, 23, 22, 20, 18]	[20, 10, 14, 21, 17, 19]
Armon Clark	[14, 15, 12, 20, 18, 16]	[19, 18, 14, 10, 15, 17]
Taylor Collins	[19, 13, 10, 14, 18, 17]	[14, 19, 15, 18, 20, 17]
Alfred Harmon	[16, 15, 20, 17, 18, 19]	[14, 15, 16, 17, 21, 20]
Martice Jackson	[13, 14, 16, 12, 18, 17]	[11, 10, 18, 12, 15, 17]
Vincent Langat	[18, 14, 16, 13, 17, 15]	[12, 21, 10, 18, 11, 20]
Jessica Stewart	[14, 11, 16, 12, 20, 13]	[21, 20, 16, 19, 15, 18]
Astride Tchakoua	[15, 11, 10, 14, 16, 19]	[15, 11, 16, 20, 18, 19]
Daren Washington	[19, 20, 15, 17, 16, 18]	[17, 20, 15, 19, 21, 16]
Marcus Wynn	[15, 13, 16, 17, 10, 20]	[16, 17, 18, 12, 20, 11]

Student Name: ______________________ ________ J#: ___________________

Q6 - 12 pts) Recall the problem of finding local minimum in an array (one dimension). We discussed the following pre-requisites for an array to have at least one local minimum.
(1) The array should have at least 3 elements
(2) The first two elements of the array should be decreasing and the last two elements of the array should be increasing
(3) The array should have distinct elements

In the slides, I did not give a formal pseudo code to find a local minimum (given the above three pre-requisites).
(a - 8 pts) Provide a pseudo code to find a local minimum in the array. (Note that: it is sufficient to find just one local minimum in the array).
(b - 4 pts) Create an array of 10 distinct integers that also satisfies the other two pre-requisites as listed above and show the execution of your pseudo code of (a) to find a local minimum.

Student Name: ________________________ ________ J#: ____________________

Q7 - 13 pts) Each of you are assigned a monotonically decreasing function, f(n), wherein n > 0 and *n* is an integer.

Determine the minimum value (threshold) of n for which f(n) is less than the target value assigned to you. Use the binary search approach discussed in class and also count the number of iterations needed to determine the threshold value of *n*.

Student Name	**Function, f(n)**	**Target**
Demetrius Brown	$3/n^2$	0.001
Jahelle Cato	$5/n^3$	0.0001
Nzefili Chukwuma	$2/n^3$	0.0004
Armon Clark	$7/n^4$	0.00001
Taylor Collins	$5/n^2$	0.005
Alfred Harmon	$6/n^4$	0.000004
Martice Jackson	$3/n^4$	0.000001
Vincent Langat	$4/n^3$	0.00002
Jessica Stewart	$5/n^4$	0.000005
Astride Tchakoua	$8/n^3$	0.00007
Daren Washington	$10/n^3$	0.0005
Marcus Wynn	$9/n^4$	0.0000001