Answers are not 100%. Please text me to confirm.

Multiple Choice Questions. Each question has 1 mark.

1. Assume that the array A = [8, 4, 5, 3, 2] needs to be sorted. What are the last two numbers swapped if it is sorted with Bubble Sort?

a. 8, 4
b. 3, 2
c. 4, 3
d. 2, 5

4 8 5 3 2 → 4 5 3 2 8 → 3 4 2 5 8 → 2 3 4
4 5 8 3 2 → 4 3 5 2 8 → 3 2 4 5 8
4 5 3 8 2 → 4 3 2 5 8
4 5 3 2 8

2. Consider the following algorithm:

```
Euclid(m,n)
  while n ≠ 0 do
     r ← m mod n
     m ← n
     n ← r
  return m
```

Assume that this algorithm is started with m = 24, n = 16, i.e., Euclid (24, 16). What are the values of m, n, and r when the algorithm terminates?

a. m = 8, n = 0, r = 0
b. m = 0, n = 8, r = 0
c. m = 4, n = 0, r = 0
d. m = 0, n = 4, r = 0

r	8	0
m	16	8
n	8	0

3. Insertion Sort is a decrease and conquer algorithm.

a. True
b. False

4. Assume that you have to search for a value in an unsorted array. Which one of the following statements is true?

a. It is always faster to sort the array before searching for the value (rather than just searching in the unsorted array)
b. It is never faster to sort the array before searching for the value (rather than just searching in the unsorted array)
c. It is faster to sort the array only when the target value is a low value and will end up near the front of the array after sorting
d. It is faster to sort the array only when the target value is a high value and will end up near the end of the array after sorting

5. Consider the Binary Search shown below. Assume the input is A: [2, 4, 6, 8, 10]. How many iterations are required (i.e.: what is the final value of iter) to find the value 2 in the array?

```
algorithm: binarySearch(a[], w, x, y)
 iter ← 0
 while x ≤ y
     iter ← iter+1
     z ← floor((x+y)/2)
     if w > a[z]
         x ← z+1
     else if w < a[z]
         y ← z-1
     else
         return z
 return not found
```

a. 1
b. 3
c. 2
d. 4

6. All brute force algorithms are $O(n^2)$.

a. True
b. False

7. What type of algorithm design technique is the recursive Binary Search algorithm most appropriately classified under?

a. Decrease by one and conquer
b. Divide and conquer
c. Brute force
d. Decrease by a constant factor and conquer

8. Divide and Conquer always yields a more efficient solution than Brute Force.

a. True
b. False

9. Consider the Fake-Coin problem. What is the best case of comparisons that are needed to identify the fake coin if you have 33 coins in your input set?

a. 1
b. 5
c. 2
d. 4

10. Assume that the array A = [6, 7, 5, 1, 2] needs to be sorted. What are the last two numbers swapped if it is sorted with Selection Sort?

6 7 5 1 2
1 7 5 6 2
1 2 5 6 7

a. 6, 7
b. 2, 7
c. 6, 1
d. 1, 5

11. Determine the number of character comparisons made when searching for pattern P in text T using the brute force technique.

T=TOPOTOP
P=POP

a. 6
b. 8
c. 5
d. 7

12. What is the efficiency of the following function?

```
public static int f(int n)
   if (n == 1) return 0;
   return 1+ f(n/2);
```

35

a. $O(n^2)$
b. O(n)
c. O(nlogn)
d. O(logn)

13. What is the efficiency of the following function (f1)?

```
public static int f1(int n) {
   if (n == 0) return 0;
   return f1(n/2) + f2(n) + f1(n/2);
}

public static int f2(int n) {
  int x = 0;
  for (int i = 0; i < n; i++)
    x++;
  return x;}
```

0 1
1 2
n-1 n

a. $O(n^2)$
b. O(n)
c. O(nlogn)
d. O(logn)

14. What is the efficiency of the following code fragment?

```
int count = 0;
int n = a.length;
Arrays.sort(a);
for (int i = 0; i < n; i++) {
  for (int j = i+1; j < n; j++) {
    if (Arrays.binarySearch(a, a[i] + a[j]))
      count++;
  }
}
```

a. $O(n^2 \log n)$
b. $O(n^2)$
c. $O(n \log n)$
d. $O(\log n)$

15. What are the correct intermediate steps of the following data set when it is being sorted with the Insertion sort? [16, 21, 11, 19]

a. 16,21,11,19 -- 11,16,21,19 -- 11,16,19,21
b. 16,19,11,21 -- 11,19,16,21 -- 11,16,19,21 -- 11,16,19,21
c. 16,11,21,19 -- 16,11,19,21 -- 11,16,19,21
d. 11,21,16,19 -- 11,16,21,19 -- 11,16,19,21

16. How many possible solutions are there for an instance of the assignment problem of size n = 4?

a. 24
b. 12
c. 60
d. 120

17. Consider the following pseudocode for an algorithm:

```
1. ALGORITHM Mystery(A[0..n-1, 0..n-1])
2. //Input: A matrix A[0..n-1, 0..n-1] of real numbers
3. for i ← 0 to n-2 do
4.    for j ← i+1 to n-1 do
5.       if A[i, j] ≠ A[j, i]
6.          return false
7. return true
```

What does this algorithm do?

a. Checks whether all the elements in the array A are equal or not.
b. Checks whether the array A is symmetrical about its main diagonal or not.
c. Checks whether all the elements on the main diagonal are equal or not.
d. Checks whether all the elements that are not on the main diagonal, are equal or not.

18. The efficiency of which one of the following sorting algorithms is not $O(n^2)$ in the worst case?

a. Insertion sort
b. Selection sort
c. Merge sort
d. Bubble sort

19. Consider the problem of computing min and max in an unsorted array where min and max are minimum and maximum elements of the array. Algorithm A1 can compute min and max in a1 comparisons with divide and conquer. Algorithm A2 can compute min and max in a2 comparisons by scanning the array linearly. What could be the relation between a1 and a2 considering the worst case scenarios?

a. Depends on the input
b. a1 < a2
c. a1 > a2
d. a1 = a2

MergeSort → nlogn

Brute-Force → 2n
+2

20. General purpose algorithms that make use of presorting can never have a worst-case efficiency that is better than O(nlogn).

a. True
b. False

nlogn
nlogn

nlogn

n + nlogn

n^2

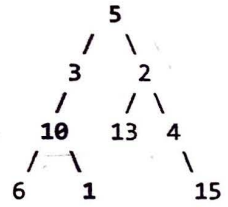
Long answer question.

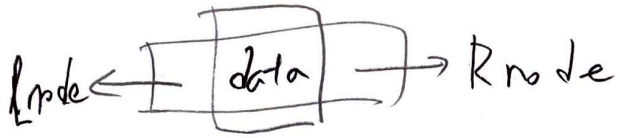
Design a **divide and conquer** **(recursive)** algorithm for the following problem. [5 marks]

Given a binary tree and a sum, return true if the tree has a root-to-leaf path such that adding up all the values along the path equals the given sum. Return false if no such path can be found.

Note: A leaf is a node with no children.

Example: Given the below binary tree and sum = 19,





return true, as there exists a root-to-leaf path 5→3→10→1 for which the sum is 19.

```
ALGORITHM FindSumPath(Head node, int sum) {
    if (sum = node.data and node.left = null and node.right = null
      return true;
    else if (sum < 0 or node = null)
      return false;
    else if (node.left = null and node.right = null)
      return false;

    if (FindSumPath(node.left, sum - node.data))
      return true;
    if (FindSumPath(node.right, sum - node.data))
      return true;

    return false;
}
```

ALGORITHMS

Student Test Report On Midt1 A

Course #: FD-COMP3760

Course Title: MIDTERM

Day/Time: 10/22

Instructor: FARNAZ DARGAHI

Description:

Term/Year: 201930

Student Name: DONGSUN, KIM

Student ID: 001052311 **Code:**

	Possible Pts.	Raw	Objective	Exam#/Essay	Percent
Midt1	20.00	15.00	15.00		75.00%

Response Description:

<dash> correct <#> multiple marks <space> no response

<alphabet> student's incorrect response <*> bonus test item

Test Items:	1-5	6-10	11-15	16-20						
Test Key.	B,A,A,B,C	B,D,B,A,B	D,D,C,A,A	A,B,C,D,A						
Answers	-,-,B,C,-	-,-,-,-,-	-,-,B,C,-	-,-,-,C,-						

Remarks: