Q1 Academic Honesty

0 Points

It is a violation of the Academic Integrity Code to look at any reference material other than your textbook and lecture notes, or to give inappropriate help to someone or to receive unauthorized aid by someone in person or electronically via messaging apps such as WhatsApp. Academic Integrity is expected of all students of Hacettepe University at all times, whether in the presence or absence of members of the faculty. Do NOT sign nor take this exam if you do not agree with the honor code.

Understanding this, I declare I shall not give, use or receive unauthorized aid in this examination.

Signature (Specify your name and surname as your signature)

Mehmet Taha USTA MTUSTA

Q2 Complexity

10 Points

Q2.1

2 Points

What is the complexity of the following code fragment in big-O notation?

```
for (int i = n; i > 0; i /= 2)
   count++;
```

- ○ $O(n)$
- ○ $O(n/2)$
- ◉ $O(\log_2 n)$
- ○ $O(n \log_2 n)$

Q2.2

2 Points

What is the big-O notation of the following complexity?

$n^3 + 10^6 n^2$

- (●) $O(n^3)$
- ( ) $O(n^2)$
- ( ) $O(10^6 n^2)$
- ( ) $O(10^6)$

Q2.3

2 Points

What is the complexity of the following code fragment in big-O notation?

```
for (i=0; i<n; i++){
    for (j=0; j<=n-1; j++){
        for (k=j; k<=n-1; k++){
            … loop body…
        }
    }
}
```

- ( ) $O(n^2)$
- (●) $O(n^2 logn)$
- ( ) $O(n^3)$
- ( ) $O(n^2/2)$

Q2.4

2 Points

What is the big-O notation of the following complexity?

$n^k + n + n^k logn$

- ○ $O(n^k)$
- ◉ $O(n^k logn)$
- ○ $O(n^k + n)$
- ○ $O(n^k + n^k logn)$

Q2.5
2 Points

What is the complexity of the following code fragment in big-O notation?

```
for (i=0; i<n; i++){
      for (j=1; j<=n-1; j*2){
      … loop body …
      }
   }
```

- ○ $O(n^2)$
- ○ $O(n^2 logn)$
- ◉ $O(nlogn)$
- ○ $O(logn)$

Q3 Recursion
10 Points

Write the output of the following code.

```
int RecursiveFunc (int array[], int index, int n)
{
        int val1, val2;
        if ( n==1 )
                return array[index];
        val1 = RecursiveFunc (array, index, n/2);
        val2 = RecursiveFunc (array, index+(n/2), n-(n/2));
        if (val1 > val2)
                return val1;
        else
                return val2;
}

int main()
{
        int a[4] = {14,12,18,15,2,16,4,13};
        cout << RecursiveFunc(a,0,8) << endl;
}
```

18

Q4 Multidimensional Array

20 Points

A matrix is called symmetric if for all values of i and j satisfies $A[i][j] = A[j][i]$. A sparse matrix has at least m/2 + 1 zero values out of all m items in the matrix. Given that matrix A is sparse, symmetric, and square (NxN).

Q4.1

10 Points

Propose and describe an efficient representation to improve the space complexity compared to two-dimensional representation of matrix A.

Using Linked Lists

```
class Linkedlist{
public:
        int row;
        int column;

        int symmetric_row;
        int symmetric_column;

        int value ;
        Linkedlist* next;
```

}

Row: row value of cell not containing zero value
Column: column value of cell not containing zero value
Value: Value of the non zero element located at [row][column] (index)
Next node: Address of the next node

The matrix a is symmetrical, so keeping its symmetric coordinates in the class will prevent the creation of extra linklist.

List items can be sorted by row values first, then column values

Using linked list representation, each of the nodes that store a NON-ZERO value can be accessed quickly by traversing the linked list.

Linked list representations save space in storing the elements compared to a traditional array.

Q4.2

10 Points

Justify and show mathematically your proposed representation is space efficient.

A sparse matrix has at least m/2 + 1 zero values out of all m items in the matrix.

A sparse matrix has at most m/2 - 1 NON-ZERO values out of all m items in the matrix.

Matrix A is symmetric so half the number of non-0 values will be linked list. (m/2 - 1) / 2

Q5 Struct/Stack/Tree

12 Points

Q5.1
2 Points

Can a Structure (struct) contain a pointer to itself?

- ○ Compilation error
- ○ Runtime error
- ◉ Yes
- ○ No

Q5.2
2 Points

The `sizeof` for a struct is always equal to the sum of `sizeof` of each individual member.

- ○ True
- ◉ False

Q5.3
4 Points

Suppose that an intermixed sequence of (stack) push and pop operations are performed. The push operations push the integers 0 through 9 in order; the pop operations print out the return value. Which of the following printed sequence could not occur?

- ○ 2 5 6 7 4 8 9 3 1 0
- ○ 4 3 2 1 0 5 6 7 8 9
- ◉ 4 6 8 7 5 3 2 9 0 1
- ○ 4 3 2 1 0 9 8 7 6 5
- ○ 2 1 4 3 6 5 8 7 9 0

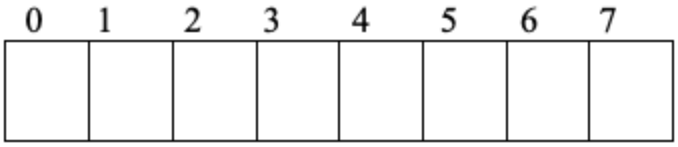
Q5.4
4 Points

What is the *maximum* height of a **full binary tree** with n nodes?

- (●) $\lceil \log_2(n+1) \rceil$
- ( ) $n/2$
- ( ) $n-1$
- ( ) n

Q6 Queue

10 Points



Enqueue(5) , Enqueue(2) , Dequeue () , Enqueue(6) , Enqueue(3) , Dequeue () , Enqueue(8) , Enqueue(2) , Enqueue(7) , Enqueue(4) , Enqueue(1)

Q6.1

5 Points

Given the initial empty position of the queue (circular) above, give the final representation of data below for array representations and fill the values of front and rear positions into an array of size 8.

Note: For your answer write each content of the array from left to right with spaces, and if there is a gap put - so if the array has empty places at index 3, 6, and 7 then a sample output will be:
1 2 3 - 4 5 - -

1 - 6 3 8 2 7 4

Q6.2

5 Points

	First	Rear
Initial	-1	-1
Final		

Write the values seperated by space e.g., 1 2

2 0

Q7 Data Structure

20 Points

You have a collection of documents and each document is composed of terms such as $D = (d_1, d_2, \ldots, d_{|D|})$ is the set of documents, $T_{d_i} = (t_1, t_2, \ldots, t_{|T_{d_i}|})$ is the set of terms within document d_i, and T is the set of unique terms in the collection. You would like to find the documents that contains the given set of terms $Q = \{(t^1, t^2, \ldots, t^k) | t^i \in T, k < |T|\}$.

Example:
$D = (d_1, d_2, d_3)$ where
T_{d_1} =(computer, engineer, data, structures, analysis)
T_{d_2} =(computer, science, database)
T_{d_3} =(computer, science, AI, data, algorithms)
T =(AI, algorithms, analysis, computer, data, database, engineer, science, structures)
The query Q =(computer, data) returns (d_1, d_3).

Choose a data structure to use in order to perform the query efficiently if $|T_{d_i}| \ll |T|, 1 \leq i \leq |D|$? Explain your reasoning in detail.

The space complexity of your data structure should be less than $O(|D||T|)$, otherwise you will not get any credit.
Note that $|.|$ denotes cardinality operator i.e., the number of elements.

I choose binary search tree data structure.
class Node{
public:
string value;

```
        Node* left;
        Node* right;
}
```

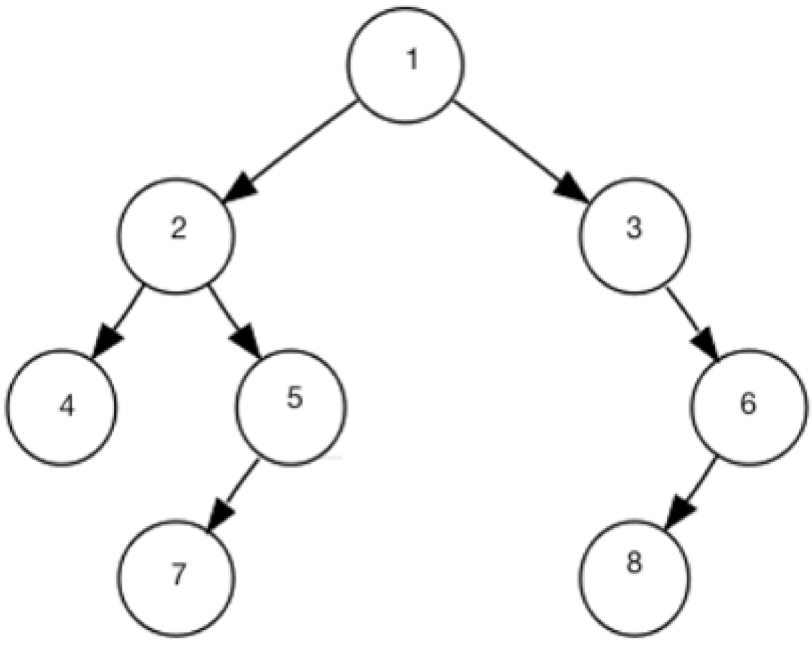
The major advantage of binary search trees over other data structures is that the related sorting algorithms and search algorithms such as in-order traversal can be very efficient.

Algorithm	Average	Worst case
Space	O(n)	O(n)
Search	O(log n)	O(n)
Insert	O(log n)	O(n)
Delete	O(log n)	O(n)

Q8 Binary Tree

18 Points

For each of the below traversal algorithms, give the order of the nodes that are visited for the below tree structure.
Write the solution as a space-separated string e.g., 1 2 3 4 5 6 7 8



Q8.1

6 Points

Preorder:

1 2 4 5 7 3 6 8

Q8.2
6 Points

Inorder:

4 2 7 5 1 3 8 6

Q8.3
6 Points

Postorder:

4 7 5 2 8 6 3 1

Midterm Exam

GRADED

STUDENT
MEHMET TAHA USTA

TOTAL POINTS
68 / 100 pts

QUESTION 1
Academic Honesty **0** / 0 pts

QUESTION 2
Complexity **8** / 10 pts

2.1 (no title) **2** / 2 pts

2.2 (no title) **2** / 2 pts

2.3 (no title) **0** / 2 pts

2.4 (no title) **2** / 2 pts

2.5 (no title) **2** / 2 pts

QUESTION 3

Recursion **10** / 10 pts

QUESTION 4

Multidimensional Array **10** / 20 pts

4.1 (no title) **10** / 10 pts

4.2 (no title) **0** / 10 pts

QUESTION 5

Struct/Stack/Tree **12** / 12 pts

5.1 (no title) **2** / 2 pts

5.2 (no title) **2** / 2 pts

5.3 (no title) **4** / 4 pts

5.4 (no title) **4** / 4 pts

QUESTION 6

Queue **10** / 10 pts

6.1 (no title) **5** / 5 pts

6.2 (no title) **5** / 5 pts

QUESTION 7

Data Structure **0** / 20 pts

+ 20 pts Correct

+ 12 pts Partially correct

+ 8 pts Space complexity constraint is satisfied but the data structure is not efficient for the specified query.

+ 5 pts Incorrect but provides some insight.

✔ **+ 0 pts** Incorrect

+ 0 pts N/A

QUESTION 8

Binary Tree **18** / 18 pts

8.1 (no title) **6** / 6 pts

8.2 (no title) **6** / 6 pts

8.3 (no title) **6** / 6 pts