International Islamic University Chittagong

Department of Computer Science and Engineering

B. Sc. in CSE Midterm Examination, Autumn 2022

Course Code: CSE 2421 Course Title: Computer Algorithms

Total marks: 30 *Time: 90 minutes*

[Answer all the questions; in some questions, there might be options; Figures in the right hand margin indicate full marks.]

1.		**CO**	**DL**	
a)	Let **K = ((X MOD 2)+1)*2**, where **X** is the last digit of your ID. Now, find the asymptotic upper bound of the following recurrence by using master method. **T(n) = KT(n/2) + n** Verify your answer through recursion tree method.	**CO4**	N	4
b)	Compare the time complexity of the following two functions. Which one runs faster.	**CO4**	E	3

```
void Chip (int n)
{
  for (int i = 1; i <= n; i++)
    for (int j = 1; j <= i; j = j*2)
      k = k + 1;
}

void Dale (int n)
{
  for (int i = 1; i <= n; i = i*2)
    for (int j = 1; j <= i; j++)
      k = k + 1;
}
```

c)	Show that the asymptotic upper bound of the following function is not $O(n^2)$: $An^3 + Bn^2 + C$. Assume that A, B, and C are any constants. **OR** Show that the asymptotic lower bound of the following function is not $\Omega(n^2)$: $An^3 + Bn^4 + Cn^5$. Assume that A, B, and C are any constants.	**CO4**	N	3
2.				
a)	Prove that we can convert an array into a heap in **O(n)** time.	**CO4**	N	4
b)	Write down the recurrence equation for the running time of quicksort algorithm in which partition always produces a 9-to-4 proportional split. Solve the recurrence using recursion tree and show that in this case running time is **O(nlgn)**.	**CO3**	N	3

c) The operation **Heap-Delete(A, i)** deletes the item in node **i** from heap **A**. Give an implementation of **Heap-Delete** that runs in **O(lgn)** time for an n-element max-heap. **OR** Show the steps of the operation **Extract-Max()** on the following Max-Heap A = ⟨10, 9, 8, 7, 6, 5, 4, 3, 2, 1⟩.	**CO2**	**A**	**3**

3.

a) Find an optimal parenthesization of a matrix-chain product whose sequence of dimensions is ⟨7, 4, 5, 3, 3⟩. **OR** Find a Longest Common Sequence of the following two sequences using dynamic programming and show the steps. X = ⟨A, B, C, D, E, F⟩ Y = ⟨D, C, B, E, A, F, E⟩	**CO1**	**A**	**4**
b) Give a brief argument why we cannot apply the technique of dynamic programming if the problem does not have optimal substructure property.	**CO5**	**N**	**2**
c) Show the operation of the first PARTITION of the quick sort algorithm on the array A = ⟨6, 4, 8, 9, 2, 3, 7, 4, 8, 5⟩	**CO1**	**A**	**2**
d) What is the smallest value of **n** such that an algorithm whose running time is **100*(X+1)*n²** runs faster than an algorithm whose running time is **2ⁿ** on the same machine? Assume that **X** is the last digit of your ID.	**CO4**	**N**	**2**