BRAC University (Department of Computer Science and Engineering)

CSE 221 (Algorithm) for Fall 2024 Semester

Quiz 1 (Set A)

Student ID:
Section: **Full Marks: 20**
Name: **Duration: 25 minutes**

---

1. Given a program, write down the time complexity. (6 marks)

```
int i, j, k, a, b, sum
for ( i = 0; i < n; i = i + 3)
      for ( j = n; j >= 1; j = j / 5)
            for ( k = 1; k <= n; k = k * 5)
                  sum = a + b
```

2. Given a program, write down the time complexity. (5 marks)

```
for (i = n / 2; i > 1; i /= 6) {
    for (j = 2; j <= i; j *= 4) {
        for (k = 0; k <= j; k *= 3) {
            p = p + n / 2;
        }
    }
}
```

3. Consider an array containing **N** unique values where for some index **i**, the values are in increasing order from index **0** to (**i-1**), and then again from **i** to **(N-1)**. Moreover, it is guaranteed that all the values from index **0** to (**i-1**) are greater than all the values from **i** to **(N-1)**.
An example array is given below.

index	0	1	2	3	4	5	6	7
value	9	12	15	2	4	5	7	8

Here **i=3**, it means the values are in increasing order from index 0 to 2, and then again from 3 to 7. Also, all values from index 0 to 2 are greater than all values from 3 to 7 (it is guaranteed, no checking required).

a. Given such an array, propose an algorithm to find the index **i**. **Write** your algorithm with a code/pseudocode/flowchart/step-by-step instructions. 6

b. **Write** the time complexity of your algorithm. 3

BRAC University (Department of Computer Science and Engineering)

CSE 221 (Algorithm) for Fall 2024 Semester

Quiz 1 (Set B)

Student ID:
Section: **Full Marks: 20**
Name: **Duration: 25 minutes**

---

1. Given a program, write down the time complexity (6 marks)

```
for (i=0; i<n; i+=4) {
    for (j=1; j<n; j*=2) {
        for (k=0; k<30; k++) {
            print("Am I still not 30?!!");
        }
        print("Why, God, why? We had a Deal!");
        for (m=n; m>0; m-=2) {
            print("Could you BE more dramatic?");
        }
    }
}
```

2. Given a program, write down the time complexity. (5 marks)

```
for (i = n / 2; i > 1; i /= 6) {
    for (j = 2; j <= i; j *= 4) {
        for (k = 0; k <= j; k *= 3) {
            p = p + n / 2;
        }
    }
}
```

3. You are given an array containing N distinct integers in a wave-like sequence. Meaning, the numbers in the beginning are in ascending order, and after a specific position, they are in descending order. For example: [1, 3, 4, 5, 9, 6, 2, -1]

You have to find the maximum number of this sequence. Can you devise an efficient algorithm such that the time complexity will be less than O(N)?

1. Present your solution idea as a pseudocode/ python code/ flowchart/ step-by-step instructions/ logical explanation in one-two paragraphs. 6
2. Write the time complexity of your algorithm. 3