

Time complexity Problem sheets:

1.

```
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.

$$f_1(n) = (\log n)^{2023}$$

$$f_2(n) = n^2 \log_n(n^n)$$

$$f_3(n) = n^3 + 7n^2$$

$$f_4(n) = 2.023^n$$

$$f_5(n) = n \log n$$

$$f_6(n) = n * \sqrt[3]{n^2}$$

Now do the followings:

- Write a correct asymptotic upper bound for each of the above.
- Sort the functions in ascending order of their growth rate, assuming n is significantly large. Just write the sorted order, no need to show any simulation.

3. Find the time-complexity of the following task in terms of number of students.

You are given a student attendance sheet. Each student has a unique integer ID. You have to count the number of students having an even number as ID. The list is sorted but the IDs are not necessarily consecutive. So you check each ID one by one.

4. Write the asymptotic time complexity of the following code snippet. Show your works/reasoning.

```
for i in range (1,n)  
    j= 1  
    while j < i*i  
        j=j+1
```

5. Write a sample code such that it's complexity can be said as

0. $O(\max(m^2, n^3))$
1. $O(m * n * \sqrt{n})$
2. $O(n * \log_{10}(n))$

6. In the primary scholarship exam in Bangladesh, four lakh ($n=4,00,000$) students take part but only the top **50** students are given an award.

Write the asymptotic time complexity to give the awards. Assume that each award is given in a constant time.

```
7. def contains_duplicates(elements):  
    for outer in range(len(elements)):  
        for inner in range(len(elements)):  
            if outer == inner:  
                continue  
  
            if elements[outer] == elements[inner]:  
                return True
```

return False

Find out worst case time complexity.

8. Is $f(n) = O(g(n))$ or is $g(n) = O(f(n))$? **Answer** for the following scenario. **Show** your calculations.

- i. $f(n) = 4^n$, $g(n) = 16^{\log_2(n)}$
- ii. $f(n) = (\sqrt{n} + n)\sqrt{n}$, $g(n) = n^2$

9. Show that $(n^2 + 5n)\log n / (n + 6) = \Theta(n \log n)$

10.

```
worstCase(n):  
    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
```

1. $n^2 \log(n)$ **2.** $f_1 < f_5 < f_6 < f_3 = f_2 < f_4$ **3.** If number of students is N , then **$O(N)$**

4. $T(N) = 1^2 + 2^2 + 3^2 + \dots + N^2 = N(N+1)(2N+1)/6$
= $O(N^3)$

The explanation that the second (nested) loop will run for $n*n$ times in the last iteration, can also be accepted.

5.

6. $O(1)$ **7. $O(n^2)$**

8.

9.

10. $O(n \log^2 n)$