

Name: _____
 Section: _____

Question 1

Marks = 4

Analyze the worst-case time complexity of the following algorithms. Use the appropriate asymptotic notation among O , Ω , and Θ to specify the time efficiency class of the given algorithms.

Algorithm Test1()

```
{
1. p = 0
2. i = n
3. while(i > 1)
4. {
5.     p++
6.     i = i/2
7. }
8. for(j = 1, j < p, j = j*2)
9. {
10.    Some of O(1) task // Basic Operation
11. }
```

Algorithm Enigma(n)

```
{
1  j ← 1, i ← 0
2  while (i < n)
3  {
4      // Some O(1) task
5      i ← i + j
6      j ← j + 1
7  }
```

ANSWER IN THIS BOX

Time Complexity of Algorithm Test1()

Time Complexity of Algorithm Enigma(n)

Question 2 (Use backside of this page)**Marks = 6**

Two friends Ali and Umar are working in a software house. Their project manager gives them a problem P and confines them only employ "*divide and conquer*" design paradigm for solving given problem P. Both come up with the following solutions:

Ali's Solution: He divides the given problem of size n into *three sub problems* each of size $n/2$, $n/4$, and $n/8$ respectively; and his module for combining solution of sub problems takes a time in $\Theta(n)$.

Umar's Solution: He divides the given problem of size n into 2 sub problems each of size $n/5$, and $7n/10$, and his module for combining solution of sub problems takes a time in $\Theta(n)$.

- Set up a recurrence relation for both solutions
- Solve the these recurrence relations
- Whose algorithm is efficient? Justify your answer.

ANSWER IN THIS BOX