Practice Worksheet

Ques.1 Let w(n) and A(n) denote respectively, the worst case and average case running time of an algorithm executed on an input of size n. which of the following is ALWAYS TRUE?

(a) $A(n) = \Omega(W(n))$ (b) $A(n) = \Theta(W(n))$ (c) $A(n) = O(W(n))$ (d) None of the above

Ques.2 Arrange these functions by order of growth from highest to lowest

$100*n^2$, 1000, 2^n, $10*n$, n^3, $2*n$

Ques.3 What is the time complexity of the following code fragments?

(a)
```
int fun(int n)
{
int count = 0;
for (int i = 0; i < n; i++)
        for (int j = 0; j < n; j++)
                for(k = 0; k < n; k++)
                        count += 1;
return count;
}
```

(b)
```
int a = 0;
for (i = 0; i < N; i++) {
        for (j = N; j > i; j--) {
                a = a + i + j;
        }
}
```

(c)
```
int i, j, k = 0;
for (i = n / 2; i <= n; i++)
{
        for (j = 2; j <= n; j = j * 2)
        {
                k = k + n / 2;
        }
}
```

(d)
```
int fun(int n)
{
        for (int i = 1; i <= n; i++)
        {
                for (int j = 1; j < n; j += i)
                {
                        // Some O(1) task
                }
}       }
```

(e)
```
void fun()
{
int i, j, count = 0;
for ( i = n/2; i <= n; i++)
        for ( j = 1; j <= n; j = j * 2)
                for( k = 1; k <= n; k = k * 2)
                        count++;
}
```

(f)
```
void fun(int n, int k)
{
    for (int i=1; i<=n; i++)
    {
            int p = pow(i, k);
            for (int j=1; j<=p; j++)
            {
                    // Some O(1) work
            }
    }
}
```

(g)
```
fun(int n)
{
     for(i = 1; i <= n; i = i*2)
     {
          for(j = 1; j <= i; j = j*2)
               printf(" Hii ");
     }
}
```

(h)
```
void fun(int n, int arr[])
{
    int i = 0, j = 0;
    for(; i < n; ++i)
       while(j < n && arr[i] < arr[j])
          j++;
}
```

(i)
```
void function(int n)
{
   int count = 0;
    for (int i=n/2; i<=n; i++)
      for (int j=1; j+n/2<=n; j = j++)
         for (int k=1; k<=n; k = k * 2)
            count++;
}
```

Ques4. For the functions, n^k and c^n, what is the asymptotic relationship between these functions? Assume that k >= 1 and c > 1 are constants.

Ques5. Decide whether these statements are True or False:

1. If f(n) = Θ(g(n)) and g(n) = Θ(h(n)), then h(n) = Θ(f(n))
2. If f(n) = O(g(n)) and g(n) = O(h(n)), then h(n) = Ω(f(n))
3. If f(n) = O(g(n)) and g(n) = O(f(n)) then f(n) = g(n)
4. $\frac{n}{100}$ = Ω(n)

Ques6. Find the complexity of below recurrence:

$$T(n) = \begin{cases} 1, & n = 0 \\ 3T(n-1), & n > 0 \end{cases}$$

Ques7. Find the complexity of below recurrence:

$$T(n) = \begin{cases} 1, & n = 0 \\ 2T(n-1) - 1, & n > 0 \end{cases}$$

Ques8. Find the complexity of below recurrence:

$$T(n) = \begin{cases} 1, & n = 0 \\ 7T(n/2) + 3n^2 + 2, & n > 0 \end{cases}$$