Practice Questions on Time Complexity

Question 1:

```
{
  for(i = 0; i < n; i++){
    // O(1)
  }
  for(j = m; j > 0; j--){
    // O(1)
  }
}
```

O(n) + O(m) = O(m+n)

Question 2:

```
for(i = 0; i < n; i += 3){
  for(j = 0; j < m; j += 2){
    // O(1)
  }
}
```

i=0, i=1, ... n/3 times

m/2 + m/2 + ... (n/3 times) = m/2 × n/3 = mn/6

= O(mn)

Question 3:

```
for(i = 1; i < n; ){
  i *= 2;
}
```

1, 2, 4, 8, ... 2^k $2^k < n$

$k < \log n$ O(log n)

Question 4:

```
for(i = 0; i < n; i++){
  for(j = 0; j < m; j++){
    k = m * n;
    while(k > 0){
      k /= 2;
    }
  }
}
```

log(m×n) × m × log(m×n) = O(n × m × log(m×n))

Question 5:

```
{
  for(i = 0; i < n; i++){
    for(j = 1; j < n; j *= 2){
      // O(1)
    }
  }
  for(i = 0; i < 100; i++){
    // O(1)
  }
}
```

log n × n = n log n = O(n log n)

100 × O(1) = O(1)

T.C: O(n log n)

Question 6:

```
for(i = n; i > 0; i /= 2){
  for(j = 0; j < i; ++j){
    // O(1)
  }
}
```

i = n, n/2, n/4, ...

n + n/2 + n/4 + ... Inf G.P

Sum of = $a\left(\frac{1}{1-r}\right)$

$= 1\left(\frac{1}{1-\frac{1}{2}}\right)$

$= 2$

$n\left(1 + \frac{1}{2} + \frac{1}{4} \cdots\right) = 2n$

T.C = O(n)

Question 7:

```
{
  j = 0;
  for(i = 0; i < n; i++){
    while(j < n && arr[i] < arr[j]) j++;
  }
}
```

→ j not getting reset

→ n + n

T.C : O(n)

Question 8:

```
{
  for(i = 0; i < n; i++){
    for(j = 0; j < m; j++){
      // O(1)
    }
  }

  for(i = 0; i < n; i++){
    for(j = 0; j < m; j /= 2){
      // O(1)
    }
  }
}
```

m + m + ... n times = m × n T.C : O(mn)

m > log m

log m + log m + .. n times = n × log m

T.C O(n log m)

O(mn) + O(n log m)

T.C O(mn)