Assignment Solutions | Time and space complexity analysis | Week 8

1. Calculate the time complexity for the following code snippet.

```
for(int i = 0; i < n; i++) {
  for(int j = 0; j * j < n; j++) {
    cout << "PhysicsWallah ";
  }
}
```

Solution :

`O(n * sqrt(n))`

2. Calculate the time complexity for the following code snippet.

```
int c = 0;
for(int i = 0; i < n; i++) {
  for(int j = 1; j < n; j *= 2) {
    c++;
  }
}
```

Solution :

`O(n log n) as the first loop 'i' will be iterated n times and the inner loop will only traverse logn times so in total the overall time complexity becomes O(nlogn).`

3. Calculate the time complexity for the following code snippet.

```
int c = 0;
for(int i = 0; i < n; i++) {
  for(int j = 1; j * j < n; j *= 2) {
    c++;
  }
}
```

Solution :

`Let us analyze how many times the inner loop will iterate. Let us see the values of j for that.`

`J = 1, 2, 4, … 2k`

`So 2^k * 2^k < n`

`So 2^(k+1) < n`

`So Time complexity becomes logN.`

4. Calculate the time complexity for the following code snippet.

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

Solution :

`Here the inner loop will be traversed ‘i’ times so let us see the values of ‘i’ here.`

`Values of ‘i’ will be n, n/2, n/4, n/8 and so on`

`So the total number of iterations in the above nested loop will be n + n/2 + n/4 + n/8 + ..`

`Which sums to 2n`

`So time complexity becomes O(2n) ~ O(n)`

5. Calculate the time complexity for the following code snippet.

```
int c = 0;
for(int i = 1; i < n; i*=2) {
  for(int j = n; j > i; j--) {
    c++;
  }
}
```

Solution :

`Lets us calculate the number of iterations in the above nested loop here, we get`

`Values of ‘i’ will be 1,2,4,8, 2^k`

`So the total number of iterations will be`

`(n-1) + (n-2) + (n-4) + .. + (n-2^k)`

`This sum becomes n*k - (1+2+4+ .. + 2^k)`

`Which becomes n*k - (2^(k+1))`

`Here k is number of terms which is O(logN)`

`Hence the overall time complexity becomes nlogn − n`

`~ O(nlogn)`