

Summary DSA

Sessions No 03(30-11-2022)

- **Asymptotic Analysis:** As a developer we have to analyze functions in terms of performance means how much time it will take for performing functions. And for denote we used some notation called Big O notation.
- **Arithmetic Progression:** It is a sequence of numbers in order, in which the difference between any two consecutive numbers is a constant value.
Eg. 1 2 3 4 5 6 7 so on.
- **Natural numbers** are the numbers that start from 1 and end at infinity.
- We have to do sum of natural numbers and for solving this problem there is two way ie.
 - 1) By human being
 - 2) By computer : Very fast compared to human being

```
int sumOfNaturalNumber(int n) {  
    int sum = 0;  
    for (int i = 1; i <= n; i++) {  
        sum = sum + i;  
    }  
    return sum;  
}
```

- Above function is one of the approach and it will take time as per n value. It totally depends on user input. Here time complexity is $O(n)$

Eg. If we provide $n=4$ then the CPU will take 4 unit time means for loop will run 4 times and 1 operation will take 1 unit time.

- Another approach to solve of above problem ie. using formula.

Sum of Natural Number = $n \times (n + 1) / 2$

Eg. If we want sum of till 5 number then,

$$5 \times (5 + 1) / 2 = 30 / 2 = 15 \text{ ie. } 1 + 2 + 3 + 4 + 5 = 15$$

Code:

```
int sumOfNaturalNumber(int n) {  
    return n * (n + 1) / 2;  
}
```

- In above code we have only 1 instruction ie. 1 operation and here it will take only 1 unit CPU time to run. Above code doesn't depend on user input. Here time complexity is $O(1)$ ie. constant
- **Search Algorithm:** Searching Algorithms are designed to check for an element or retrieve an element from any data structure where it is stored.
- As human beings we can see all data at one point in time ie. we can see entire data in one go.
- Computers can only see one data at one point in time. If we have multiple data and if we want to search particular data then the computer searches one by one. And for search we have to create algorithm.

Eg. data = [5, 2, 4, 9, 8, 1, 3] and if we want to search 4.

Algorithm:

- 1) Pick first data ie. 5 and compare with search data ie. 4
If equal then stop here if not equal then go ahead.
- 2) Pick second data ie. 2 and compare with search data ie. 4
If equal then stop here if not equal then go ahead.
- 3) Pick third data ie. 4 and compare with search data ie. 4

If we compare both data then we can see they are equal and we can stop here.

This above algorithm is known as **pseudo code**.

- If we get an answer in the first find then this scenario is called **Best(Lucky) Case Scenario**. And for this, time complexity is $O(1)$.
- If we got the answer at last then this scenario is called **Worst Case Scenario**. And for this, time complexity is $O(n)$.