Assignment Day 4 | 28th December 2020

Question-1:

In the Binary Search algorithm, it is suggested to calculate the mid as beg + (end - beg) / 2 instead of (beg + end) / 2. Why is it so?

Answer:

In the Binary search ‘(beg+end)/2’ is subject to overflow with large values. With iteration it’s not valid.

In ‘beg + (end - beg) / 2’ it avoids the overflow and this also works with iterators, as the outcome for (end-beg) is a number. As we can subtract two iterators to get the distance between them, but we can’t add two iterators.

So that’s why beg + (end - beg) / 2 is better and most used and effective method than (beg+end)/2.

Question-2:

Write the algorithm/function for Ternary Search.

Function for ternary search:

```
int Ternary_Search (int beginning, int end, int data, int array [])
{
        while(end>=1)
        {
         int mid1=beginning + (end-beginning)/3;
         int mid2 = end – (end-beginning)/3;

         if(array[mid1] == data)
          {
                return mid1;
          }

          if(array[mid2] == data)
          {
                return mid2;
          }
```

```
      if(data < array[mid1])
      {
          return Ternary_Search (beginning, mid1-1, data, array);
      }
      else if (data > array[mid2])
      {
          return Ternary_Search (end, mid2+1, data, array);
      }
      else
      {
          return Ternary_Search (mid1+1, mid2-1, data, array);
      }
      }
    return -1;
}
```