ASSIGNMENT-2

1. Use divide-and-conquer technique to calculate sum of an integer array. Use two different ways to define subproblems. For each way:

a. Write the pseudocode of the algorithm.

→ Solution:-

```
sumArray (a[], start, end)
INT mid, lsum, rsum                        ---- 1
IF end == start
    RETURN a[start]                        ---- 1
ELSE
    mid = (end + start)/2                  ---- 1
    lsum = sumArray(a, start, mid)         ---- n/2
    rsum = sumArray(a, mid+1, end)         ---- n/2
RETURN lsum + rsum                         ---- 1
```

b. c. ~~solution~~ Give the running time recurrence (recursive equation) and calculate running time in θ.

→ $T(n) = 2T(n/2) + 1$

Using Master's theorem,

$T(n) = aT(n/b) + f(n)$

where $a \geq 1$; $b > 1$ and $f(n) = \theta(n^k \log^p n)$

$\therefore a = 2$, $b = 2$, $f(n) = \theta(1) = \theta(n^0 \log^0 n)$

$\therefore k = 0$, $p = 0$

Case 1:- $\log_b a > k \Rightarrow \log_2 2 > k \Rightarrow 1 > 0 \Rightarrow$ TRUE

$\therefore \theta(n^{\log_b a}) = \theta(n^{\log_2 2}) = \theta(n^1) = \theta(n)$

Solution 2:-

```
sumArray2(a[]):
IF a == NULL
    RETURN 0
IF a.length == 1
    RETURN a[0]
FOR(INT i=01; i<a.length; i++)
    RETURN a[0] + sumArray2(a[i])
```

Running time recurrence:-

$T(n) = T(n-1) + 1$

Using ~~Master~~ substitution theorem,

as $T(n) = T(n-1) + f(n)$ .... where $f(n) = 1$

$= T(n-1) + 1$

$\Rightarrow T(n-1) = T((n-1)-1) + 1 = T(n-2) + 1$

$\therefore T(n) = T(n-2) + 1 + 1 = T(n-2) + 2$

$\therefore T(n) = T(n-k) + k$

For $k = n-1$, $T(n-k) = T(1)$

$\therefore T(n) = T(1) + n - 1$

$T(n) = \Theta(n)$

2. Use divide-and-conquer technique to search a number in the sorted list of n numbers.

a. Write the pseudo code of the algorithm.

→

```
searchNumber(a[], n, key)
  INT left = 0
  INT right = n-1
  WHILE left <= right                  ......... O(1)
    INT mid = (left + right)/2
    IF a[mid] == key                   ......... O(1)
      RETURN mid
    IF a[mid] > key                    ......... T(n/2)
      right = mid - 1
    ELSE
      left = mid + 1
```

RETURN -1

b. Write the recursive running time equation (recurrence). with result.

→ $T(n) = T(n/2) + 2 * \theta(1)$

$= T(n/2) + \cancel{2}C = T(n/2)$

Using Masters theorem.

$a = 1,\ b = 2,\ n = 1 \quad f(n) = \theta(1) = \theta(n^0 \log^0 n)$

$\therefore k = 0,\ p = 0$

Case 1:- $\log_b a > k \Rightarrow \log_2 1 > k \Rightarrow 0 > 0 \Rightarrow$ FALSE

Case 2:- $\log_b a = k \Rightarrow \log_2 1 = 0 \Rightarrow 0 = 0 \Rightarrow$ TRUE

i) $p > -1 \Rightarrow 0 > -1 \Rightarrow$ TRUE

$\therefore \theta(n^k \log_2^{p+1} n) = \theta(n^0 \log_2^{0+1} n) = \theta(\log_2 n)$