Arpita Singh

2310071005

Experiment Task 1

1 — Algorithm (Divide & conquer)

```
merge and Count (arr, temp-arr, left, mid, right)
Initialise i,j,k and inversion_count to 0
  If arr[i] <= arr[j]
      temp-arr [k] = arr [i]
      i++
  Else
    temp-arr [k] = arr [j]
    Increment inv_sount by (mid - i + 1)
    j ++
 k++
   copy remaining element of left subarray, if any
   copy remaining element of right subarray, if any
Return inversion count.

merge-sort_and_count (arr, temp_arr, left, right)
If left < right
  Find mid = ((left + right)/2
  Count left inv = merge sort_and_count (arr, tem_arr, left, mid)
  count right inv = merge_sort_and_count (arr, temp_arr, mid+1, right)
  count split inv = merge and Count (arr, temp_arr, left, mid, right)
  Return (left + right + split)
```

— Algorithm (Brute force)

```
Initialize inv_count to 0
 For i from 0 to n-1:
    For j from i+1 to n:
    If arr[i] > arr[j], increament inv_count
Return inv_count
```

231071005

Test Case

2] Positive test Cases

① Input : 165370
Expected output : 8

② Input : 373708
Expected output : 5

③ Input : 592290
Expected output : 9

④ Input : 686641
Expected output : 11

⑤ Input : 827581
Expected output : 9

Negative test Cases

① Input : 123abc
Expected: Output: Error (invalid input)

② Input : 5
Expected output : Error (single digit input)

③ Input : 111111
expected output: 0 as all digits are same

④ Input: -123567
Expected output: Error (as negative input)

⑤ Input : 1234
Expected output : Error(course code should be six digit only)

3] Time Complexity

A] Brute force Algorithm :

Outer loop , loop runs n times , ~~for~~
where n is no. of elements
Inner loop runs n-i times

when i=0 , inner loop runs n-1 times
when i=1 , inner loop runs n-2 times
⋮
when i=n-1 , inner loop runs 0 times

Total $(n-1)+(n-2)+(n-3)+ \ldots 1+0 = \frac{n(n-1)}{2} \approx \frac{n^2}{2} = O(n^2)$

∴, Brute force time complexity $O(n^2)$

B] Divide and Conquer Algorithm

merge step takes linear time O(n) for merging two halves
- Each level of recursion processes n element
- height of recursion tree is log n

Total time complexity

$$T(n) = 2T\left(\frac{n}{2}\right) + O(n)$$

From masters theorem
$a=2$, $b=2$, $f(n) = O(n)$ $n^{\log_2 2} = n$
∴, $T(n) = O(n\log n)$
and also it is piggy back on merge sort $O(n\log n)$

Arpita Singh

·231071005

Experiment Task - 2

1) - Brute force Algorithm

```
// Input : Two integers x and y
Initialization : Initialize variable result to 0
Loop:
      For each digit in y :
   • Multiply x by current digit of y (from least significant
                                       to most significant)
   • Shift the result acc to position of
      the current digit in y (using power of 10)
   • Add the shifted result to result
return result
//Output : result of multiplication
```

- Karatsuba Algorithm (Divide and conquer)

```
//Input : Two integer x and y
 Base Case : If either number has only one digit, return
                                                   product
Split : For two n digit number, split in 2 halves
    a and b are high and low part of x
    c and d are high and low part of y
        $x = a \cdot 10^m + b$ , $y = c \cdot 10^m + d$   ($m = \lceil n/2 \rceil$)
Recursive calls :
    • Compute a.c ($Z_0$ = karatsuba(a,c))
    • compute b.d ($z_1$ = karatsuba (b,d))
    • compute (a+b).(c+d) ($z_2$ = karatsuba(a+b,c+d))
Combine results :
  • use formula
     result = $z_0 10^{2m} + (z_2 - z_1 - z_0) \cdot 10^m + z_1$
 return result
```

23107I005

Test Cases

2)

Positive test cases

① Input : (123, 456)
Expected output : 56088

② Input : (12, 34)
Expected output : 408

③ Input : (0, 1237)
Expected output : 0

④ Input : (721, 314)a
Expected output : 226394

⑤ Input : (43, 67)
Expected output : 2881

Negative Test Cases

① Input : (None, 100)
Expected output : Error (None is not valid input type)

② Input : (12, 'abc')
Expected output : Error (Invalid input type)

③ Input : (10.5, 2)
Expected output : Error (Floating-point numbers are invalid)

④ Input : (-123, 456)
Expected output : -56088 (or error (if neg multipication not allowed))

⑤ Input : (0, 0)
Expected output : 0 (not error but multiplication with 0)

23107100 5

Time Complexity - Task 2

3]

A) Brute force algorithm

outer loop: iterates over each digit of y

inner loop: $O(n)$ for multiplying with single digit

$O(n)$ for shifting

Total Cost is $O(n) + O(n) = O(n)$

outer loop runs n times

Each iteration takes $O(n)$

Total time complexity =

$$T(n) = n \times O(n) = O(n^2)$$

so brute force time complexity is $\underline{\underline{O(n^2)}}$

B) Divide and conquer - karatSuba

$$T(n) = 3T\left(\frac{n}{2}\right) + O(n)$$

$3T(n/2)$ as three recursive multiplication of half-sized number

$O(n)$ for comibgning of result

Masters theorem -

$a = 3$ $b = 2$

so $n^{\log_b a} = n^{\log_2 3} = n^{1.585}$

∴, time complexity of karatsuba algorithm is $O(n^{\log_2 3}) \approx \underline{\underline{O(n^{1.585})}}$