3.d. 4-G-Array Sum Max Problem

Aim:

Given an array of N integer, we have to maximize the sum of arr[i] * i, where i is the index of the element (i = 0, 1, 2, ..., N).Write an algorithm based on Greedy technique with a Complexity O(nlogn).

Input Format:

First line specifies the number of elements-n

The next n lines contain the array elements.

Output Format:

Maximum Array Sum to be printed.

Sample Input:

5

2 5 3 4 0

Sample output:

40

Algorithm:

```
function main() {
    initialize n // number of elements
    read n from user

    initialize arr array of size n // array to hold integers

    // read values into the arr array
    for i from 0 to n-1 {
        read arr[i] from user
    }

    // sorting the array using bubble sort
```

```
    for i from 0 to n-2 {
        for j from 0 to n-i-2 {
            if arr[j] is greater than arr[j+1] {
                // swap arr[j] and arr[j+1]
                initialize temp as arr[j]
                arr[j] = arr[j+1]
                arr[j+1] = temp
            }
        }
    }

    initialize prod to 0 // variable to hold the weighted sum

    // compute the weighted sum
    for i from 0 to n-1 {
        prod = prod + (arr[i] * i) // accumulate the weighted sum
    }

    print prod // output the final result
}
```

Program:

```
#include<stdio.h>
int main(){
    int n;
    scanf("%d",&n);
    int arr[n];
    for(int i=0;i<n;i++){
        scanf("%d",&arr[i]);
```

```
    }
    for(int i=0;i<n-1;i++){ for(int
        j=0;j<n-i-1;j++){
            if(arr[j]>arr[j+1]){
                int temp=arr[j];
                arr[j]=arr[j+1];
                arr[j+1]=temp;
            }
        }
    }
    int prod=0;
    for(int i=0;i<n;i++){
        prod+=(arr[i]*i);
    }
    printf("%d",prod);
}
```

Output:

	Input	Expected	Got	
✔	5 2 5 3 4 0	40	40	✔
✔	10 2 2 2 4 4 3 3 5 5 5	191	191	✔
✔	2 45 3	45	45	✔