

Merge Sort

CODE:

```
#include <stdio.h>

void merge(int arr[], int left, int mid, int right) {
    int n1 = mid - left + 1;
    int n2 = right - mid;
    int L[n1], R[n2];
    for (int i = 0; i < n1; i++) L[i] = arr[left + i];
    for (int j = 0; j < n2; j++) R[j] = arr[mid + 1 + j];
    int i = 0, j = 0, k = left;
    while (i < n1 && j < n2)
        arr[k++] = (L[i] <= R[j]) ? L[i++] : R[j++];
    while (i < n1) arr[k++] = L[i++];
    while (j < n2) arr[k++] = R[j++];
}

void mergeSort(int arr[], int left, int right) {
    if (left < right) {
        int mid = (left + right) / 2;
        mergeSort(arr, left, mid);
        mergeSort(arr, mid + 1, right);
        merge(arr, left, mid, right);
    }
}

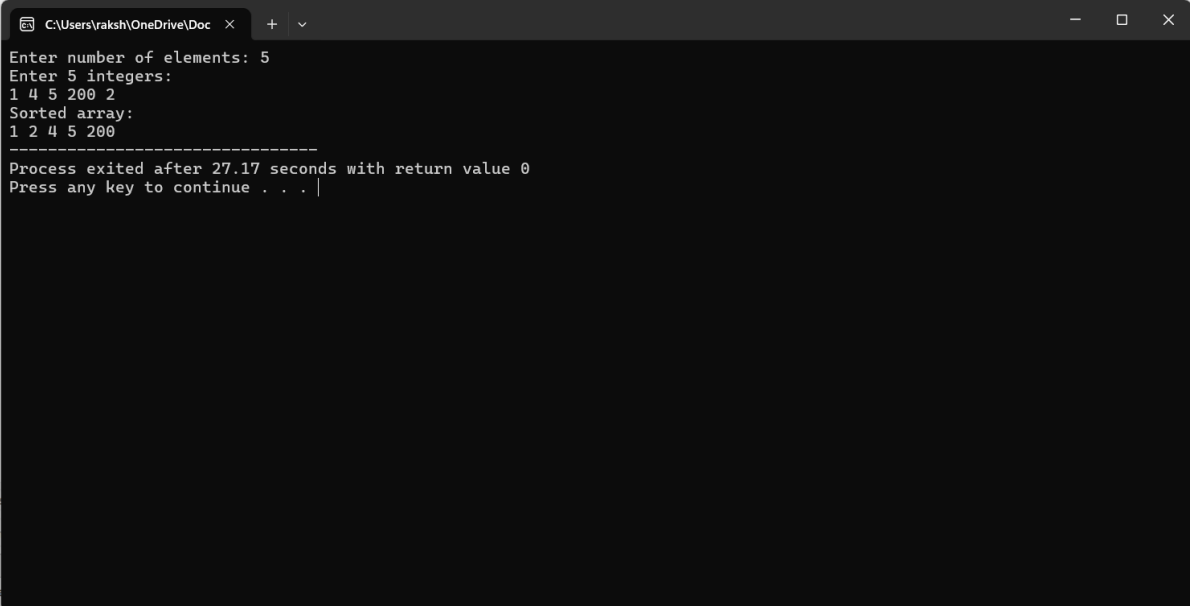
int main() {
    int arr[100], n;

    printf("Enter number of elements: ");
    scanf("%d", &n);

    printf("Enter %d integers:\n", n);
```

```
    for (int i = 0; i < n; i++)  
        scanf("%d", &arr[i]);  
  
    mergeSort(arr, 0, n - 1);  
  
    printf("Sorted array:\n");  
    for (int i = 0; i < n; i++)  
        printf("%d ", arr[i]);  
  
    return 0;  
}
```

OUTPUT:



```
C:\Users\raksh\OneDrive\Doc x + v  
Enter number of elements: 5  
Enter 5 integers:  
1 4 5 200 2  
Sorted array:  
1 2 4 5 200  
-----  
Process exited after 27.17 seconds with return value 0  
Press any key to continue . . . |
```