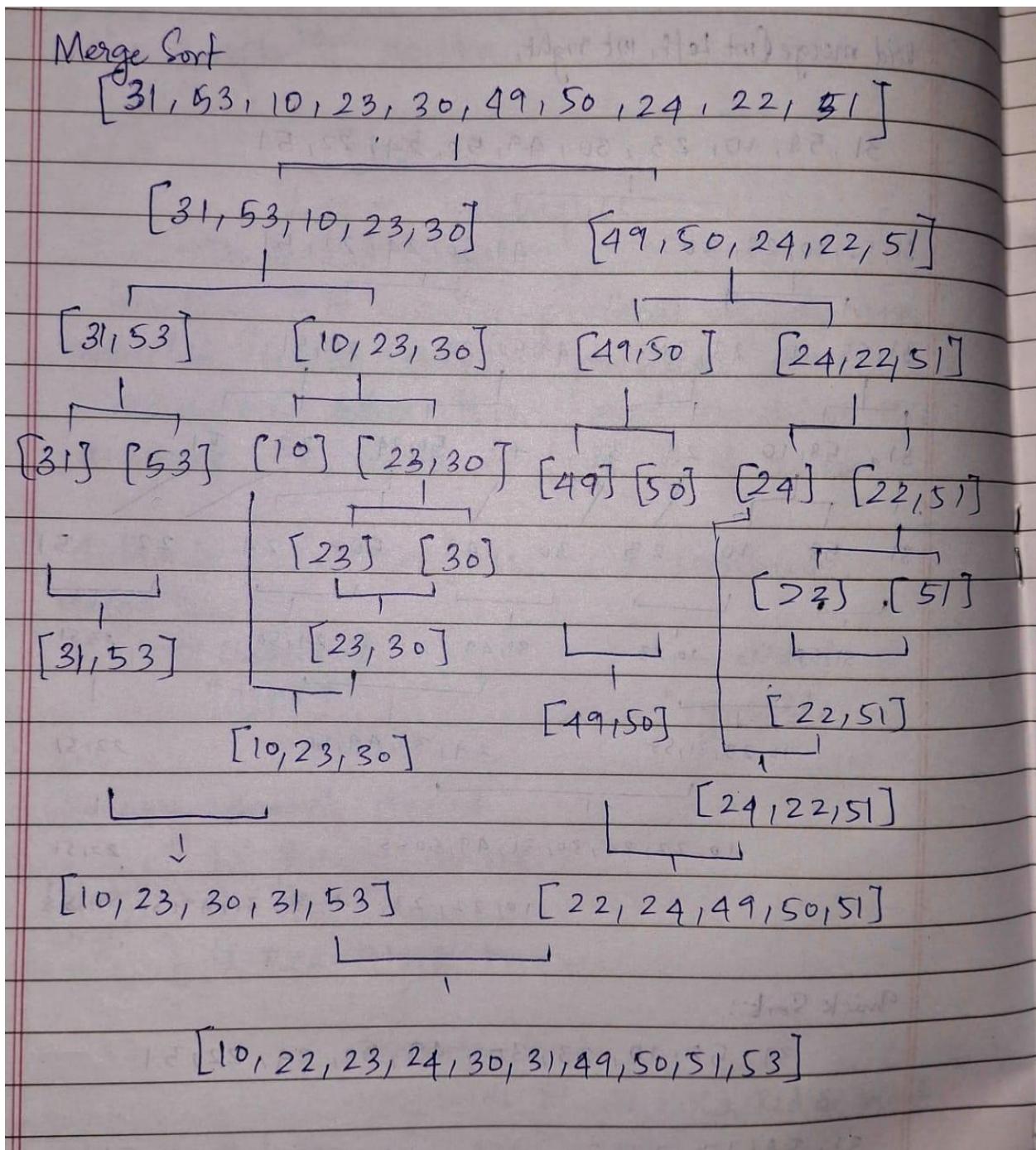


# DAA ASSIGNMENT 5

## 1. Merge sort

Method:



Code:

```
#include <stdio.h>
void merge(int arr[], int low, int mid, int high) {
    int i = low, j = mid + 1, k = 0;
    int temp[100];

    while (i <= mid && j <= high) {
        if (arr[i] < arr[j])
            temp[k++] = arr[i++];
        else
            temp[k++] = arr[j++];
    }

    while (i <= mid)
        temp[k++] = arr[i++];

    while (j <= high)
        temp[k++] = arr[j++];

    for (i = low, k = 0; i <= high; i++, k++)
        arr[i] = temp[k];
}

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

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

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

printf("Enter %d elements:\n", n);
for (i = 0; i < n; i++)
    scanf("%d", &arr[i]);

mergeSort(arr, 0, n - 1);

printf("Sorted array:\n");
for (i = 0; i < n; i++)
    printf("%d ", arr[i]);

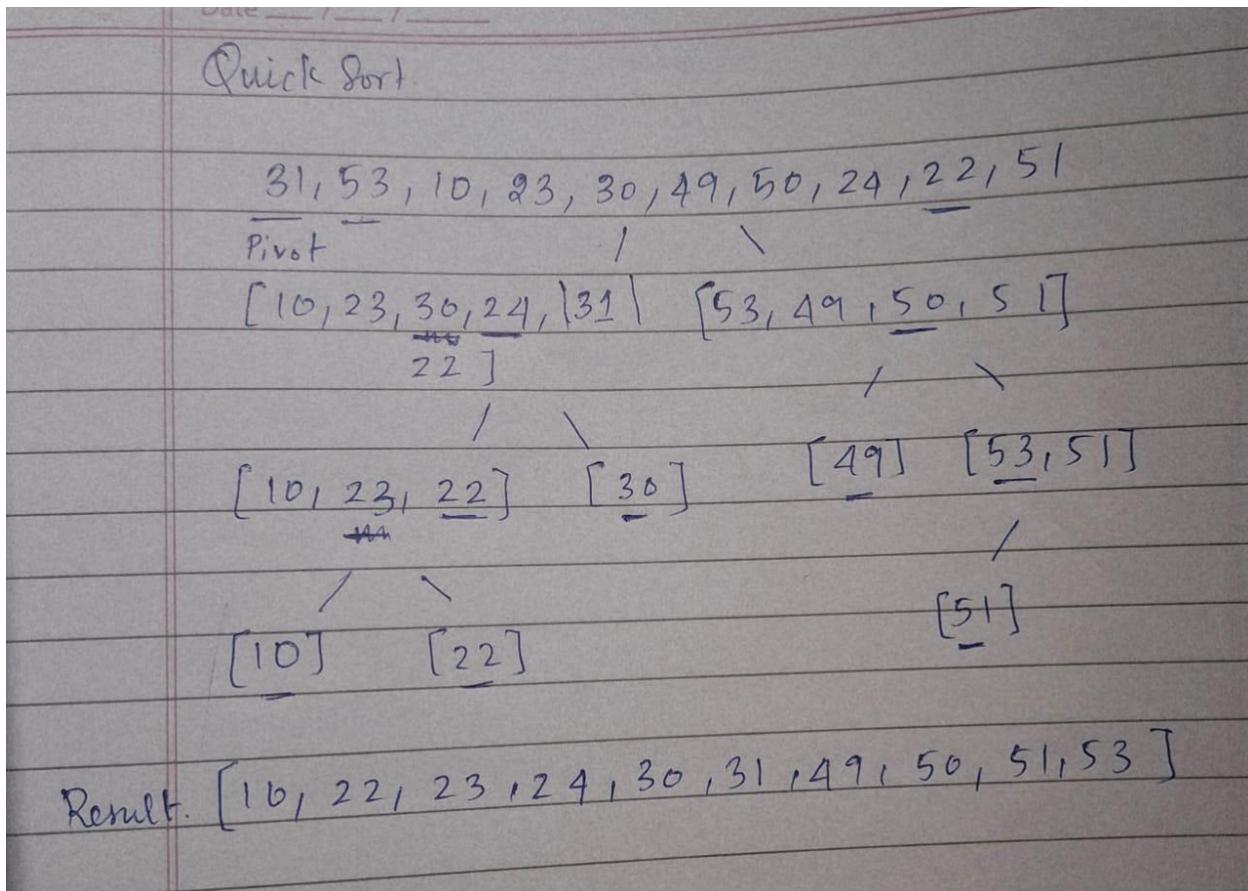
return 0;
}
```

Output:

```
Enter number of elements: 10
Enter 10 elements:
31
53
10
23
30
49
50
24
22
51
Sorted array:
10 22 23 24 30 31 49 50 51 53
```

## 2. Quick sort:

Method:



Code:

```
#include <stdio.h>
#include <stdlib.h>
#include <time.h>
void swap(int *a, int *b) {
    int temp = *a;
    *a = *b;
    *b = temp;
}
int partition(int arr[], int low, int high) {
    int randomIndex = low + rand() % (high - low + 1);
    swap(&arr[randomIndex], &arr[high]);
    int pivot = arr[high];
```

```
int i = low - 1;

for (int j = low; j < high; j++) {
    if (arr[j] <= pivot) {
        i++;
        swap(&arr[i], &arr[j]);
    }
}
swap(&arr[i + 1], &arr[high]);
return i + 1;
}

void quickSort(int arr[], int low, int high) {
    if (low < high) {
        int pos = partition(arr, low, high);
        quickSort(arr, low, pos - 1);
        quickSort(arr, pos + 1, high);
    }
}

int main() {
    int n, arr[100];
    printf("Enter number of elements: ");
    scanf("%d", &n);
    printf("Enter %d elements:\n", n);
    for (int i = 0; i < n; i++)
        scanf("%d", &arr[i]);
    srand(time(NULL));
    quickSort(arr, 0, n - 1);

    printf("Sorted array:\n");
    for (int i = 0; i < n; i++)
        printf("%d ", arr[i]);

    return 0;
}
```

Output:

```
pooja@DESKTOP-5DBTBML:~/quicksort$ ./quicksort
Enter number of elements: 10
Enter 10 elements:
31 53 10 23 30 49 50 24 22 51
Sorted array:
10 22 23 24 30 31 49 50 51 53
```