MERGESORT

Aim

Write a C Program to merge the elements of the two ordered arrays and form a single array sorted in ascending order.

1 Mergesort Algorithm

1.1 Algorithm

```
Step 1: initailize a1[100],a2[100],a3[100]
Step 2: n1=Enter elements in list 1
Step 3: n2=Enter elements in list 2
Step 4: Enter the elements of first list in sorted way
    for(i=0 to n1)
        a1[i]=Input number
    end for
Step 5: Enter the elements of second list in sorted way
    for(i=0;i<n2;i++)
        a2[i]=Input number
    end for
Step 6: Set i=0,j=0,k=0
Step 7: While(i<n1 and j<n2)
        if (a1[i]<a2[j])
            a3[k]=a1[i]
            i++
            k++
        else if (a1[i]>a2[j])
        a3[k]=a2[j]
        j++
        k++
        else
        a3[k]=a1[i]
        a3[k+1]=a2[j]
        i++
        j++
        k=k+2
        end if
    end while
Step 8: if (i==n1)
        While(a2[j]!='\0')
        a3[k]=a2[j]
        k++,j++
        end while
        else if (j==n2)
        while(a1[i]!='\0')
        a3[k]=a1[i]
        k++,i++
        end while
        end if
Step 9: set i=0
Step 10: while(i<k-2)
            Print a3[i]
            i++
        end while
```

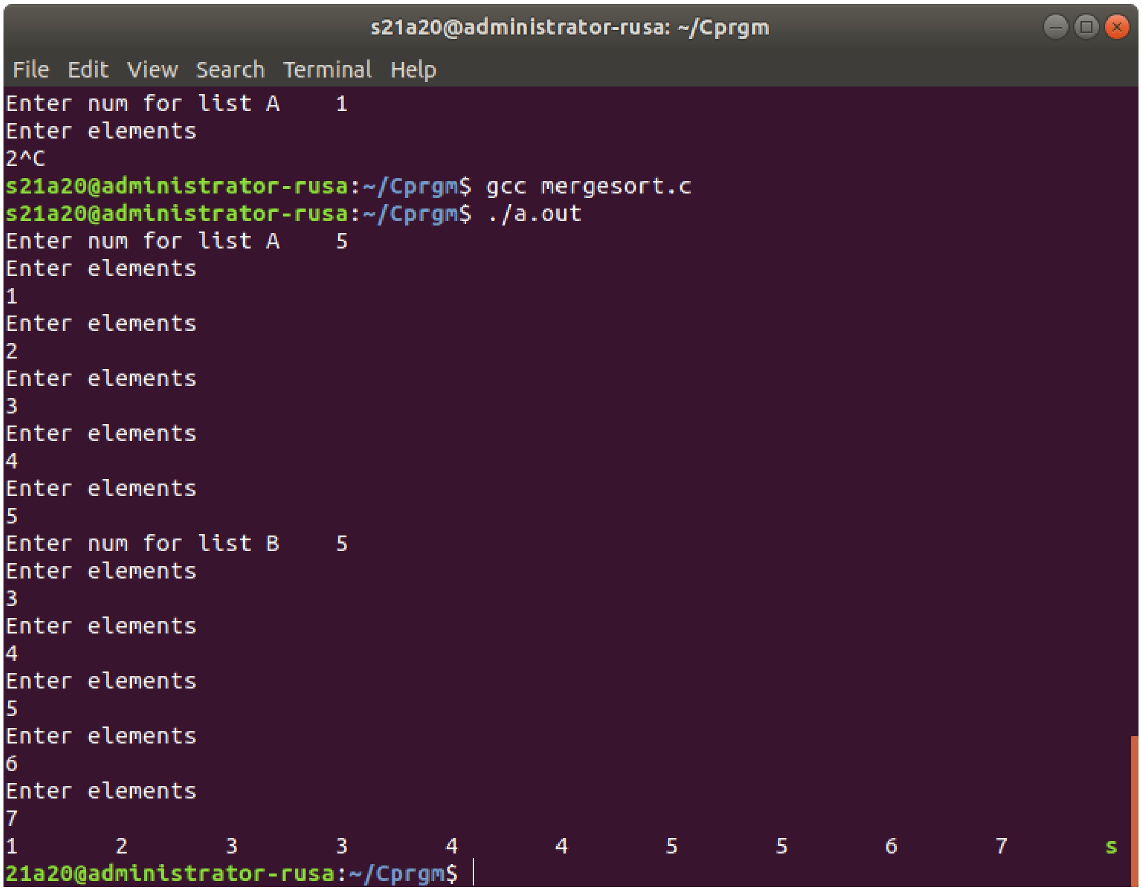
1.2 Program

```
#include <stdio.h>
void main()
{
    int i, j, k = 0, n1, n2, A[10], B[10], M[50], t = 0;
    printf("Enter num for list A \t");
    scanf("%d", &n1);
    for (i = 0; i < n1; i++)
    {
        printf("Enter elements\n");
        scanf("%d", &A[i]);
    }

    printf("Enter num for list B \t");
    scanf("%d", &n2);
    for (i = 0; i < n2; i++)
    {
        printf("Enter elements\n");
        scanf("%d", &B[i]);
    }
    i = 0, j = 0, k = 0;
    while (i < n1 && j < n2)
    {
        if (A[i] <= B[j])
        {
            M[k] = A[i];
            i++;
            k++;
        }
        else
        {
            M[k] = B[j];
            k++;
            j++;
        }
    }
    if (i == n1)
    {
        while (j < n2)
        {
            M[k] = B[j];
            j++;
            k++;
        }
    }
    else if (j == n2)
    {
        while (i < n1)
        {
            M[k] = A[i];
            i++;
            k++;
        }
    }

    for (i = 0; i < k; i++)
    {
        printf("%d\t", M[i]);
    }
}
```

1.3 Sample Input and Output


s21a20@administrator-rusa: ~/Cprgm
File Edit View Search Terminal Help

```
Enter num for list A    1
Enter elements
2^C
s21a20@administrator-rusa:~/Cprgm$ gcc mergesort.c
s21a20@administrator-rusa:~/Cprgm$ ./a.out
Enter num for list A    5
Enter elements
1
Enter elements
2
Enter elements
3
Enter elements
4
Enter elements
5
Enter num for list B    5
Enter elements
3
Enter elements
4
Enter elements
5
Enter elements
6
Enter elements
7
1       2       3       3       4       4       5       5       6       7       s
21a20@administrator-rusa:~/Cprgm$
```

Figure 1: Output

1.4 Result

Implemented Mergesort by reading two sorted arrays and printing a single sorted array.