MERGE SORT

Sri venkateswara college of engineering

Submitted by:
A.Ram Teja Reddy
22BFA04200

Abstract

Efficient Merging

- The primary goal is to combine two pre-sorted arrays. This creates a new array. It contains all elements in sorted order.

Algorithm Overview

- We'll use a two-pointer approach. It minimizes comparisons. This results in optimal performance and efficient use of memory.

Algorithm: Two-Pointer Approach

1

Initialization

Initialize pointers for each input array. Set another pointer for the output array.

2

Comparison

Compare elements at each pointer. Copy the smaller element to the output array.

3

Increment

Increment the pointer of the array that provided the smaller element.

4

Repeat

Repeat until all elements are copied to the output array.

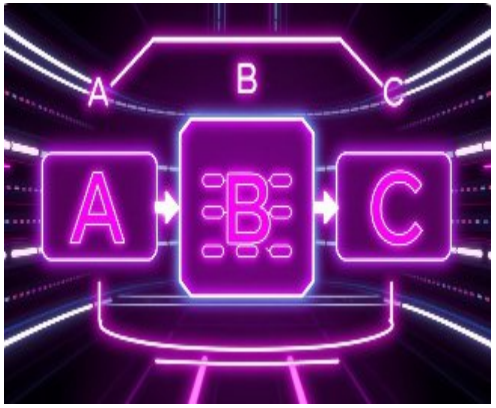
Pictorial Representation

Comparison

Visualization of comparing elements at current pointers.

Element Movement

Showing the smaller element being copied to the merged array.



Pointer Update

Illustrating the pointer increment after each copy operation.

Test Cases

Test Case 1:

- arr1: {1, 3, 5}
- arr2: {2, 4, 6}
- Expected Output: {1, 2, 3, 4, 5, 6}

Test Case 2:

- arr1: {1, 2, 3}
- arr2: {4, 5, 6}
- Expected Output: {1, 2, 3, 4, 5, 6}

Test Case 3:

- arr1: {1, 5, 9}
- arr2: {2, 3, 4, 6}
- Expected Output: {1, 2, 3, 4, 5, 6, 9}

C Code Implementation

```
#include <stdio.h>
int main()
{
    int n1,n2,n3;
    int a[10000], b[10000], c[20000];
    printf("Enter the size of first array: ");
    scanf("%d",&n1);
    printf("Enter the array elements: ");
    for(int i = 0; i < n1; i++)
       scanf("%d", &a[i]);
    printf("Enter the size of second array: ");
        scanf("%d",&n2);
    printf("Enter the array elements: ");
    for(int i = 0; i < n2; i++)
       scanf("%d", &b[i]);
```

```
n3 = n1 + n2;
    for(int i = 0; i < n1; i++)
       c[i] = a[i];
    for(int i = 0; i < n2; i++)
        c[i + n1] = b[i];

    printf("The merged array: ");
    for(int i = 0; i < n3; i++)
        printf("%d ", c[i]);
    printf("\nFinal array after sorting: ");
    for(int i = 0; i < n3; i++){
        int temp;
        for(int j = i + 1; j < n3; j++) {
            if(c[i] > c[j]) {
                temp = c[i];
                c[i] = c[j];
                c[j] = temp;
            } } }
    for(int i = 0; i < n3 ; i++)
        printf(" %d ",c[i]);
    return 0;
}
```

Output

Enter the size of first array: 5

Enter the array elements: 10 20 30 40 50

Enter the size of second array: 5

Enter the array elements: 96 73 52 48 17

The merged array:

10 20 30 40 50 96 73 52 48 17

Final array after sorting:

10 17 20 30 40 48 50 52 73 96


Thank
you