19BIT0292

Bhaumik Tandan

DIGITAL ASSIGNMENT-3

DATA STRUCTURES AND ALGORITHMS LABORATORY

CSE2011

L57+L58

Q1) Write a program to illustrate the operation of merge sort on the array A = {3,41, 52, 26, 38, 57, 9, 49, 08, 15, 72}. The output must show step by step evaluation of the algorithm.

(a) sort.h

CODE

```
void disp()
{
   printf("\nThe array is: ");
   for(int i=0;i<n;i++)
   printf("%d ",arr[i]);
}

void merge(int l, int m, int r)
{
   int l1=m-l+1,l2=r-m;

   int left[l1],right[l2];

   for(int i=0;i<l1;i++)
   left[i]=arr[i+l];

   for(int i=0;i<l2;i++)
   right[i]=arr[i+m+1];

   int a1=0,a2=0,f=l;

   while(a1<=m-l && a2<r-m)
   if(left[a1]<right[a2])
   arr[f++]=left[a1++];
   else
   arr[f++]=right[a2++];

   while(a1<l1)
   arr[f++]=left[a1++];

   while(a2<l2)
   arr[f++]=right[a2++];
}
```

```
void mergeSort(int l, int r)
{
    if (l <r)
    {
        int m = l + (r - l) / 2;
        mergeSort(l, m);
        mergeSort(m + 1, r);
        merge(l, m, r);
        disp();
    }
}
```

(b) merge_sort.c

```c
#include <stdio.h>

int arr[]={3,41, 52, 26, 38, 57, 9, 49, 8, 15, 72};
int n=sizeof(arr)/sizeof(int);

#include "sort.h"

main()
{
    mergeSort(0,n-1);
}
    q->n=t;
    swap_next(h,q->n);
}
```

OUTPUT

```
The array is: 3 41 52 26 38 57 9 49 8 15 72
The array is: 3 41 52 26 38 57 9 49 8 15 72
The array is: 3 41 52 26 38 57 9 49 8 15 72
The array is: 3 41 52 26 38 57 9 49 8 15 72
The array is: 3 26 38 41 52 57 9 49 8 15 72
The array is: 3 26 38 41 52 57 9 49 8 15 72
The array is: 3 26 38 41 52 57 8 9 49 15 72
The array is: 3 26 38 41 52 57 8 9 49 15 72
The array is: 3 26 38 41 52 57 8 9 15 49 72
The array is: 3 8 9 15 26 38 41 49 52 57 72
PS C:\Users\bhaum\OneDrive\Desktop\dsa_da>
```

Q2) Write program for searching an element in a given array of elements {45, 23,89, 20, 67, 22, 19, 10, 60, 24, 90, 76, 52, 4, 98, 56}. Search an element using linear search and recursive binary search.

(a) search.h

CODE

```c
#include<stdio.h>

int arr[]= {45, 23, 89, 20, 67, 22, 19, 10, 60, 24, 90, 76, 52, 4, 98, 56};
int n=sizeof(arr)/sizeof(int);

#include "sort.h"

void linerSearch(int a)
{
    for(int i=0;i<n;i++)
    if(arr[i]==a)
    {
        printf("\nElement found at %d",i);
        return;
    }
    else
    {
        printf("\nElement not found at %d",i);
    }

    printf("\nElement not found in the array");
}

int binarySearch(int st,int en,int s)
{
    if(st+1>=en)
    return -1;
    int m=st+(en-st)/2;
    if(arr[m]==s)
    return m;
    printf("\nElement Not found at %d",m);
    if(arr[m]>s)
```

```c
      return  binarySearch(st,m,s);
    if(arr[m]<s)
    return  binarySearch(m,en,s);
}

main()
{
    printf("\nLinear search:");
    linerSearch(60);
    // for binary search array must be sorted
    printf("\n\nMerge Sort: ");
    mergeSort(0,n-1);
    printf("\n\nBinary Search: ");

    int i=binarySearch(0,n,60);
    if(i==-1)
    printf("\nElement not found");
    else
    printf("\nElement found at %d",i);
}
```

OUTPUT

```
Linear search:
Element not found at 0
Element not found at 1
Element not found at 2
Element not found at 3
Element not found at 4
Element not found at 5
Element not found at 6
Element not found at 7
Element found at 8

Merge Sort:
The array is: 23 45 89 20 67 22 19 10 60 24 90 76 52 4 98 56
The array is: 23 45 20 89 67 22 19 10 60 24 90 76 52 4 98 56
The array is: 20 23 45 89 67 22 19 10 60 24 90 76 52 4 98 56
The array is: 20 23 45 89 22 67 19 10 60 24 90 76 52 4 98 56
The array is: 20 23 45 89 22 67 10 19 60 24 90 76 52 4 98 56
The array is: 20 23 45 89 10 19 22 67 60 24 90 76 52 4 98 56
The array is: 10 19 20 22 23 45 67 89 60 24 90 76 52 4 98 56
The array is: 10 19 20 22 23 45 67 89 24 60 90 76 52 4 98 56
The array is: 10 19 20 22 23 45 67 89 24 60 76 90 52 4 98 56
The array is: 10 19 20 22 23 45 67 89 24 60 76 90 52 4 98 56
The array is: 10 19 20 22 23 45 67 89 24 60 76 90 4 52 98 56
The array is: 10 19 20 22 23 45 67 89 24 60 76 90 4 52 56 98
The array is: 10 19 20 22 23 45 67 89 24 60 76 90 4 52 56 98
The array is: 10 19 20 22 23 45 67 89 4 24 52 56 60 76 90 98
The array is: 4 10 19 20 22 23 24 45 52 56 60 67 76 89 90 98

Binary Search:
Element Not found at 8
Element Not found at 12
Element found at 10
PS C:\Users\bhaum\OneDrive\Desktop\dsa_da>
```