Date 27/5/21
Page

WRITE-UP (LAB:7) (1BM19CS158)

Merge Sort

```
#include <stdio.h>
#include <stdlib.h>
#include <time.h>
void split (int [], int, int);
void combine (int [], int, int, int);
int main()
{
int a [15000], n, i, j, ch, temp;
clock_t start, end;
while(1)
{
printf("\n1: For manual entry of N value and array elements.");
printf("\n2: To display time taken for sorting number of
elements N in the range 500 to 14500.");
printf("\n3: To exit from the program.");
printf("\n Enter your choice:");
scanf("%d", &ch);
switch(ch)
{
case 1: printf("\nEnter the number of elements:");
scanf("%d", &n);
printf("\nEnter array elements:");
for(i=0; i<n; i++)
{
    scanf("%d", &a[i]);
}
start = clock()
split (a,0,n-1);
end = clock();
printf("\nSorted array is:");
```

```
for (i=0 ; i<n; i++)
printf ("%d \t", a[i]);
printf("\n Time taken to sort %d numbers is %f secs", n,
((double)(end-start))/CLOCKS_PER_SEC));
break;
case 2: n=500;
        while (n<=14500)
        {
        for (i=0; i<n; i++)
        {
          a[i]=n-i;
        }
        start= clock();
        split (a,0,n-1);
        //Dummy Loop to create delay
        for (j=0; j< 500000; j++)
        {
          temp=38/600;
        }
        end clock();
printf("\n Time taken to sort %d numbers is %f secs", n,
((double)(end-start))/CLOCKS_PER_SEC));
        n=n+1000;
        }
break;
case 3:
        exit (0);
}
}
return 0;
}
```

Date
Page

(1BM19CS158)

```
void split( int a[], int low, int high)
{
  int mid;
  if (low < high)
  {
    mid = (low+high)/2;
    split (a, low, mid);
    split ( a, mid+1, high);
    combine (a, low, mid, high);
  }
}
void combine ( int a[], int low, int mid, int high)
{
  int c[15000], i, j, k;
    i=k=low;
    j= mid+1;
  while (i<=mid && j<=high)
    {
      if (a[i]<a[j])
        {
          c[k] = a[i];
          ++k;
          ++i;
        }
      else
      {
        c[k] = a[j];
        ++k;
        ++j;
      }
    }
```

```
if (i>mid)
{
  while (j<=high)
  {
    c[k]=a[j];
    ++k;
    ++j;
  }
}
if (j>high)
{
  while (i<=mid)
  {
    c[k]=a[i];
    ++k;
    ++i;
  }
}
for (i=low; i<=high; i++)
{
  a[i]=c[i];
}
}
```