Date :
Page No. :

LAB - 7 mergesort

```
#include<stdio.h>
#include<stdlib.h>
#include<time.h>

void merge (int arr[], int l, int m, int r)
{ int i, j, k

  int n1 = m - l + 1;
  int n2 = r - m;

int L[n1], R[n2];

for (i = 0; i < n1; i++)
{ L[i] = arr[l + i]; }

for (j = 0; j < n2; j++)
R[j] = arr[m + 1 + j];

i = 0;
j = 0
k = l

while (i < n1 && j < n2) {
if (L[i] <= R[j]) { arr[k] = L[i]; i++;
```

Date :

Page No. :

```
else
{ arr[k] = R[j];
    j++;
}
k++;
while (i < n1) {
arr[k] = L[i];
 i++;
 k++;
 }
while (j < n2) {
arr[k] = R[j];
  j++;
   k++;
  }
}
```

Date :

Page No. :

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

void printarray (int A[], int size)
{ int i;
for (i=0; i<size; i++)
  printf ("%d, A[i]);
   printf ("\n"); }
int main ()
{ int i, n;
```

Date :
Page No :

```
printf("enter no of elements :- ");
scanf("%d", &n);
int arr[n];
srand(time(0));
for(i=0; i<n; i++)
{ scanf("%d", &arr[i]);
// alt u can use arr[i] = rand(); //
}
int arr_size = sizeof(arr) / sizeof(arr[0]);
printf("Given array is \n");
printArray(arr, arr_size);
clock_t t;
t = clock();
mergeSort(arr, 0, arr_size - 1);
printf("Sorted array is \n");
printArray(arr, arr_size);
```

Date :

Page No :

```
t = clock() - t;
double time_taken = ((double)t) / CLOCKS_PER_
SEC;

printf("fun() took %f seconds to
        execute \n", time_taken);

return 0;

}
```