ADA LAB TEST - 2

S.R.POOJA
1BM19CS135
4-C
8-7-2021

→ HeapSort technique and compute its time taken.

```
#include <stdio.h>
#include <stdlib.h>
#include <time.h>
int arr[1000000];
int temp;
void maxheap(int arr[], int size, int i)
{
  int j,k;
  for (k=0; k<80; k++)
  {
   for (j=0; j<40; j++)
    {
    }
   }
int largest = i;
int left = 2*i+1;
int right = 2*i+2;
if (left < size && arr[left] > arr[largest])
 largest = left;
if (right < size && arr[right] > arr[largest])
   largest = right;
 if largest (!=i)
  {
   temp = arr[i];
   arr[i] = arr[largest];
 arr[largest] = temp;
maxheap (arr, size, largest);
```


S.R.Pooja

```
}
}
void printArray(int arr[], int n)
{
 int i;
for (i=0; i<n; i++)
 printf("%d", arr[i]);
 printf("\n");
}
int main()
{
 time_t start, end;
 int n;
 srand(time(0));
printf("Enter the no. of elements\n");
scanf("%d", &n);
for (int i = 0; i<n; i++)
 {
  arr[i] = rand();
}
 start = time(NULL);
 heapsort(arr, n);
 end = time(NULL);
 printf("The array is sorted\n");
printf("The time taken is %.10f\n", difftime(end, start)/
       CLOCKS_PER_SEC);
 return 0;
}
```


S.R.Pooja

Modification: Using Min heap, sort the given set of N integers.

Minimum heap:

```
void heap (int a[], int n)
{
int i,j,k,temp;
for (i=2; i<=n; i++)
 {
 j=i;
 k=j/2;
temp= a[j];
while (k>0 && a[k]>temp)
{
 a[j] = a[k];
 j=k;
 k= k/2;
}
 a[j]= temp;
}
}
void heap(int a[],int n)
{
int i,j,k,temp;
for (i=n/2; i>0; i--)
 {
 k=i;
 temp = a[k];
 j= 2*k;
 while (j<=n)
 {
  if (j<n && a[j] < a[j+1])
  { j = j+1;
 }
 if (temp <a[j]) {
 a [k]= a [j];
 k=j;
```


SR. Proja

```
j= 2*k;
}
else{
break;
}
}
a[k]= temp;
}
}
void adjust(int a[], int n){
for(int i=0; i<100; i++)
{
for(int i=0; i<10; i++)
{
}
}
int i=2, temp = a[1];
while(i<=n){
if (i<n && a[i]>a[i+1])
{
i=i+1;
}
if(a[i] >i*2<temp)
{
a[i/2]= a[i];
i= i*2;
}
else
{
break;
}
}
a[i/2]. temp;
}
```