ADA Lab Test-2

Harsh Mishra
1BM17CS028

Q. Sort a given set of N-integer elements using Heap sort technique and compute its time taken.

```
#include <time.h>
#include <stdio.h>
#include <stdlib.h>
void swap(int *a, int *b)
{
   int t = *a;
  *a = *b;
  *b = t;
}
void heap(int arr[], int n, int i)
{int largest = i;
 int l = 2* i+1;
int r = 2* i+2;
 if (l<n& & arr[l]>arr[largest])
      largest=l;
 if (r<n && arr[r]>arr[largest])
      largest =r;
  if (largest != i)
{
  swap(&arr[i], &arr[largest]);
   heap(arr, n, largest);
  }
}
void heapSort(int arr[], int n)
{for(int i = n/2-1; i>=0; i--)
   heap(arr, n, i);
  for(int i = n-1; i>0; i--)
  {
```

Harsh Mishra
1BM17CS028

```
swap(&arr[0], &arr[i]);
heap(arr, i, 0);
}
}
int main()
{
clock_t start, end;
double t;
for(int n = 100; n < 601; n = n+100)
{
  int array[n];
  for(int i = 0; i < n; i++)
  {
    array[i] = rand()%1000;
  }
  start = clock();
  heapSort(array, n);
  end = clock();
  t = ((double)(end - start)) / CLOCKS_PER_SEC;
  printf("\n Time taken by Heap Sort for %d elements : %lf\n", n, t);
}
}
```

Modification

```
void minheap(int arr[], int n, int i)
{ int smallest = i;
int l = 2*i+1;
int r = 2*i+2;
if (l<n && arr[l] < arr[smallest])
    smallest = l;
if (r<n && arr[r] < arr[smallest])
    smallest = r;
```

```
    if (smallest != i)
    {
        swap(arr[i], arr[smallest]);
        mainheap(arr, n, smallest);
    }
}
{ int main()
    int arr[] = {4, 6, 3, 1, 7}
    int n = size of(arr) / size of(arr[0]);
    heap sort(arr, n);
    # print the array.
```