Ada - Lab Test 2.

IBM16CS142
SADIN SHRESTHA

*) 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 *x, int *y)
{
  int t = *x;
  *x = *y;
  *y = t;
}
void head (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 (largest != i)
  if (r < n && arr[r] > arr[largest])
      largest = r;
```

```
if (largest != i)
{
swap (&arr[i], &arr[largest]);
heapify (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--)
{
  swap (&arr[0], &arr[i]);
  heap (arr, i, 0);
 }
}
int main()
 {
   clock_t start, end;
   double ti
      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)) / ~~clock~~ CLOCKS_PER_SEC;
printf("\n Time taken by HeapSort for %d element:
        %.lf\n", n,t);

}
}
```

modification:

```
void minheap (int arr[], int n, int i)
{
int smallest = i;
int l = 2*i + 1;
int l = 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]);
    minheap (arr, n, smallest);
   }
}
int main()
    int arr[] = {0 4, 6, 3, 1, 7}
    int n = size of (arr) / size of (arr[0]);
    heapsort (arr, n);
 # print the array.
```