HEAP SORT

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

void swap(long *a, long *b)
{
    int temp = *a;
    *a = *b;
    *b = temp;
}

void heapify(long arr[], int n, int i)
{
    int largest = i;
    int left = 2 * i + 1;
    int right = 2 * i + 2;

    if (left < n && arr[left] > arr[largest])
        largest = left;

    if (right < n && arr[right] > arr[largest])
        largest = right;

    if (largest != i)
    {
        swap(&arr[i], &arr[largest]);

        heapify(arr, n, largest);
    }
}

void heapSort(long arr[], int n)
{

    for (int i = n / 2 - 1; i >= 0; i--)
```

```
        heapify(arr, n, i);

    for (int i = n - 1; i > 0; i--)
    {
        swap(&arr[0], &arr[i]);
        heapify(arr, i, 0);
    }
}

void printArray(long arr[], int n)
{
    for (int i = 0; i < n; ++i)
        printf("%d ", arr[i]);
    printf("\n");
}

int main()
{

    int n;
    long arr[100000];
    printf("Enter the array size: ");
    scanf("%d", &n);
    printf("\nEnter the array elements: ");
    for (int i = 0; i < n; i++)
    {
        scanf("%d",&arr[i]);
        // arr[i] = rand() % 100000;
    }

    clock_t start = clock();
    heapSort(arr, n);
    clock_t end = clock();
    double time_taken = ((double)(end - start)) / CLOCKS_PER_SEC;

    printf("Sorted array: ");
    printArray(arr, n);
```

```
    printf("\nTime Taken: %f seconds", time_taken);
    return 0;
}
```

```
Enter the array size: 5

Enter the array elements: 5 4 3 2 1
Sorted array: 1 2 3 4 5

Time Taken: 0.000000 seconds
```