***C code for Heap sort***

***Name: Ishanya***

***Roll no: 21329***

***DSA Assignment***

/\* Name: Ishanya

Roll number: 21329

DSA assignment \*/

#include <stdio.h>

#include <stdlib.h>

#include <time.h>

void heapify(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) {

int temp = arr[i];

arr[i] = arr[largest];

arr[largest] = temp;

heapify(arr, n, largest);

}

}

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

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

heapify(arr, n, i);

for (int i = n-1; i >= 0; i--) {

int temp = arr[0];

arr[0] = arr[i];

arr[i] = temp;

heapify(arr, i, 0);

}

}

int main() {

int n = 1000000;

int arr[n];

srand(time(0));

clock\_t tStart, tEnd;

double totalTime = 0.0;

for (int i = 0; i < 100; i++) {

for (int i = 0; i < n; i++) {

arr[i] = rand() % n;

}

tStart = clock();

heapSort(arr, n);

tEnd = clock();

double timeTaken = (double)(tEnd - tStart)/CLOCKS\_PER\_SEC;

printf("Iteration %d time taken: %.6f s\n", i+1, timeTaken);

totalTime += timeTaken;

}

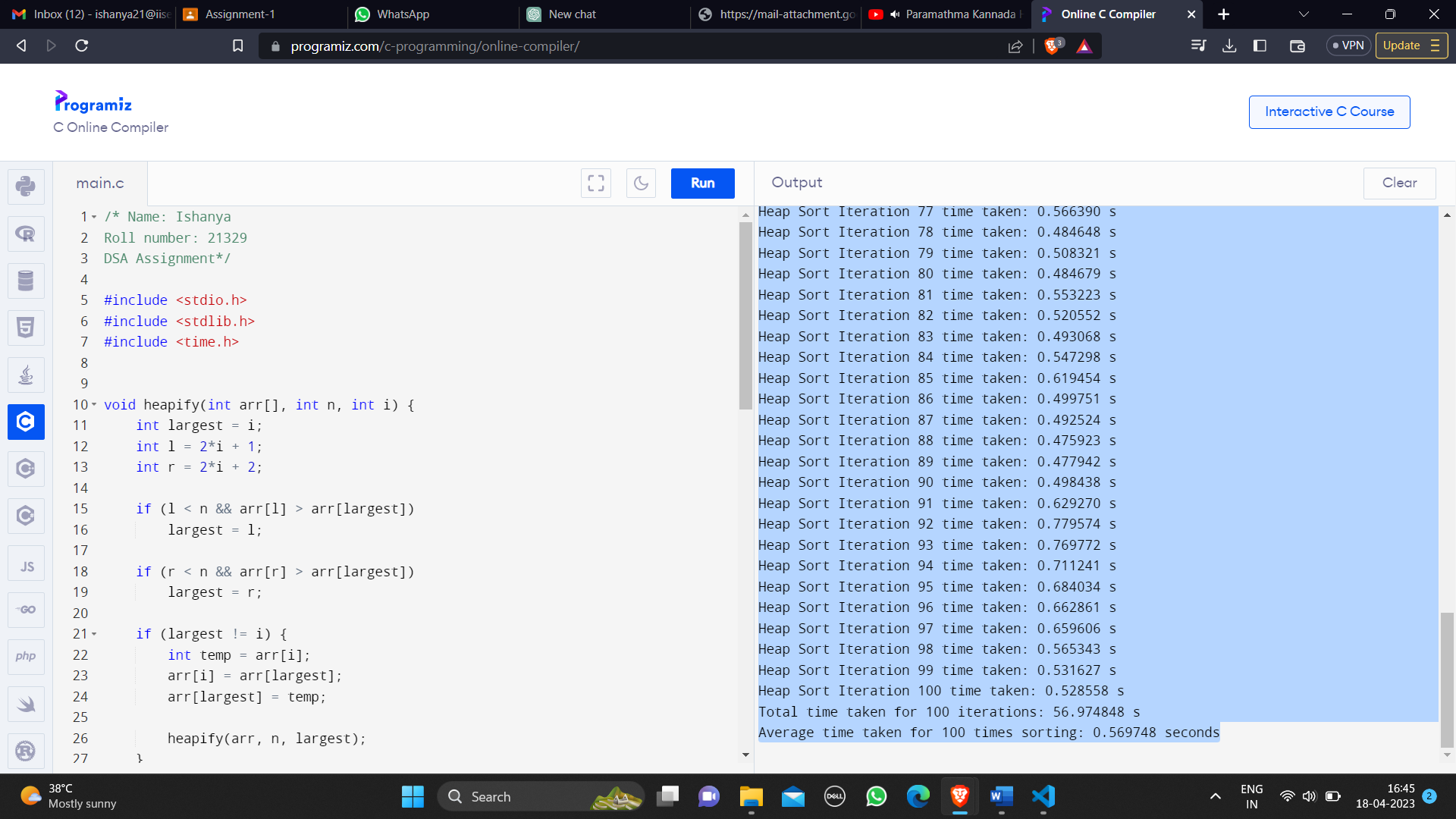
printf("Total time taken for %d iterations: %.6f s\n", 100, totalTime);

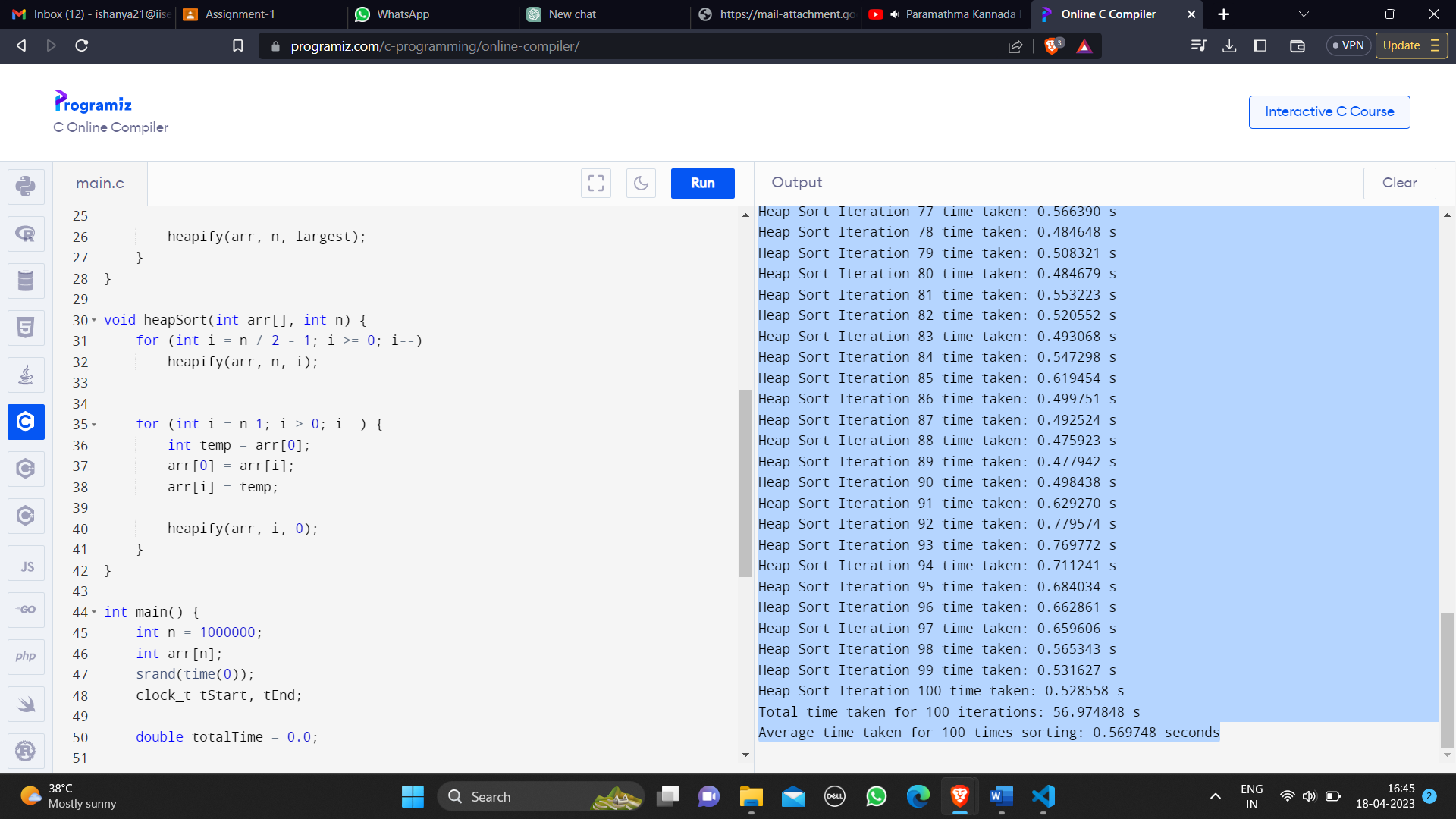
double avg\_cpu\_time\_used = totalTime / 100;

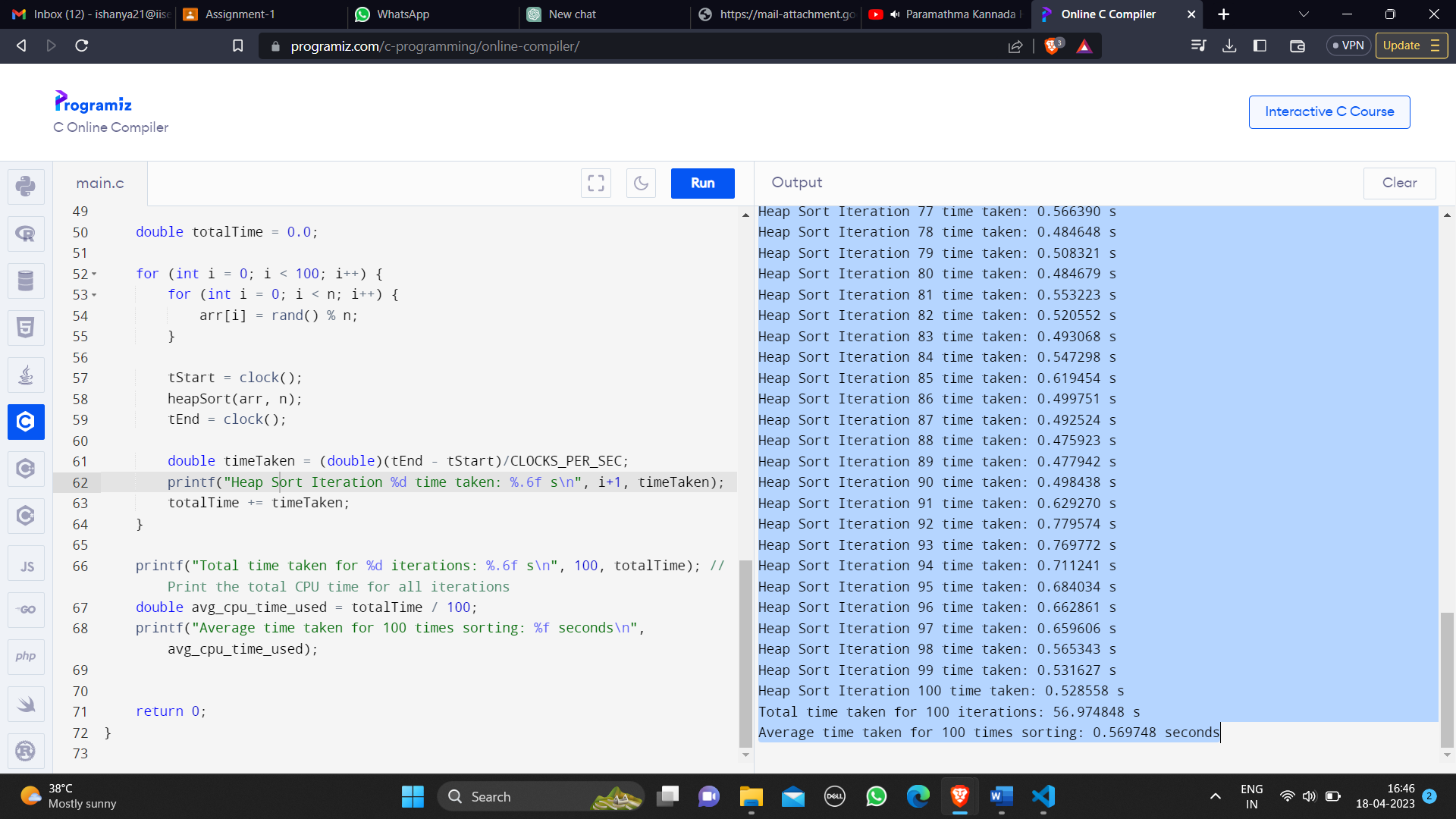
printf("Average time taken for 100 times sorting: %f seconds\n", avg\_cpu\_time\_used);

return 0;

}







***Report on the CPU time***

Heap Sort Iteration 1 time taken: 0.583616 s

Heap Sort Iteration 2 time taken: 0.535322 s

Heap Sort Iteration 3 time taken: 0.601186 s

Heap Sort Iteration 4 time taken: 0.623169 s

Heap Sort Iteration 5 time taken: 0.613100 s

Heap Sort Iteration 6 time taken: 0.681302 s

Heap Sort Iteration 7 time taken: 0.612344 s

Heap Sort Iteration 8 time taken: 0.596815 s

Heap Sort Iteration 9 time taken: 0.592889 s

Heap Sort Iteration 10 time taken: 0.560642 s

Heap Sort Iteration 11 time taken: 0.556821 s

Heap Sort Iteration 12 time taken: 0.531643 s

Heap Sort Iteration 13 time taken: 0.488827 s

Heap Sort Iteration 14 time taken: 0.478936 s

Heap Sort Iteration 15 time taken: 0.472872 s

Heap Sort Iteration 16 time taken: 0.499908 s

Heap Sort Iteration 17 time taken: 0.508630 s

Heap Sort Iteration 18 time taken: 0.657259 s

Heap Sort Iteration 19 time taken: 0.600249 s

Heap Sort Iteration 20 time taken: 0.556846 s

Heap Sort Iteration 21 time taken: 0.478952 s

Heap Sort Iteration 22 time taken: 0.494304 s

Heap Sort Iteration 23 time taken: 0.495095 s

Heap Sort Iteration 24 time taken: 0.538793 s

Heap Sort Iteration 25 time taken: 0.496032 s

Heap Sort Iteration 26 time taken: 0.537058 s

Heap Sort Iteration 27 time taken: 0.480509 s

Heap Sort Iteration 28 time taken: 0.542533 s

Heap Sort Iteration 29 time taken: 0.500980 s

Heap Sort Iteration 30 time taken: 0.534975 s

Heap Sort Iteration 31 time taken: 0.610293 s

Heap Sort Iteration 32 time taken: 0.661004 s

Heap Sort Iteration 33 time taken: 0.683067 s

Heap Sort Iteration 34 time taken: 0.776534 s

Heap Sort Iteration 35 time taken: 0.659065 s

Heap Sort Iteration 36 time taken: 0.628177 s

Heap Sort Iteration 37 time taken: 0.585408 s

Heap Sort Iteration 38 time taken: 0.592856 s

Heap Sort Iteration 39 time taken: 0.595032 s

Heap Sort Iteration 40 time taken: 0.579993 s

Heap Sort Iteration 41 time taken: 0.558260 s

Heap Sort Iteration 42 time taken: 0.590655 s

Heap Sort Iteration 43 time taken: 0.576687 s

Heap Sort Iteration 44 time taken: 0.515775 s

Heap Sort Iteration 45 time taken: 0.527021 s

Heap Sort Iteration 46 time taken: 0.564095 s

Heap Sort Iteration 47 time taken: 0.537947 s

Heap Sort Iteration 48 time taken: 0.512995 s

Heap Sort Iteration 49 time taken: 0.573570 s

Heap Sort Iteration 50 time taken: 0.552518 s

Heap Sort Iteration 51 time taken: 0.561579 s

Heap Sort Iteration 52 time taken: 0.591486 s

Heap Sort Iteration 53 time taken: 0.599167 s

Heap Sort Iteration 54 time taken: 0.530827 s

Heap Sort Iteration 55 time taken: 0.526839 s

Heap Sort Iteration 56 time taken: 0.558720 s

Heap Sort Iteration 57 time taken: 0.697626 s

Heap Sort Iteration 58 time taken: 0.661825 s

Heap Sort Iteration 59 time taken: 0.630233 s

Heap Sort Iteration 60 time taken: 0.530086 s

Heap Sort Iteration 61 time taken: 0.494131 s

Heap Sort Iteration 62 time taken: 0.509988 s

Heap Sort Iteration 63 time taken: 0.550729 s

Heap Sort Iteration 64 time taken: 0.599899 s

Heap Sort Iteration 65 time taken: 0.670962 s

Heap Sort Iteration 66 time taken: 0.495808 s

Heap Sort Iteration 67 time taken: 0.527956 s

Heap Sort Iteration 68 time taken: 0.505745 s

Heap Sort Iteration 69 time taken: 0.556702 s

Heap Sort Iteration 70 time taken: 0.600263 s

Heap Sort Iteration 71 time taken: 0.621875 s

Heap Sort Iteration 72 time taken: 0.574682 s

Heap Sort Iteration 73 time taken: 0.600044 s

Heap Sort Iteration 74 time taken: 0.616226 s

Heap Sort Iteration 75 time taken: 0.596781 s

Heap Sort Iteration 76 time taken: 0.488013 s

Heap Sort Iteration 77 time taken: 0.566390 s

Heap Sort Iteration 78 time taken: 0.484648 s

Heap Sort Iteration 79 time taken: 0.508321 s

Heap Sort Iteration 80 time taken: 0.484679 s

Heap Sort Iteration 81 time taken: 0.553223 s

Heap Sort Iteration 82 time taken: 0.520552 s

Heap Sort Iteration 83 time taken: 0.493068 s

Heap Sort Iteration 84 time taken: 0.547298 s

Heap Sort Iteration 85 time taken: 0.619454 s

Heap Sort Iteration 86 time taken: 0.499751 s

Heap Sort Iteration 87 time taken: 0.492524 s

Heap Sort Iteration 88 time taken: 0.475923 s

Heap Sort Iteration 89 time taken: 0.477942 s

Heap Sort Iteration 90 time taken: 0.498438 s

Heap Sort Iteration 91 time taken: 0.629270 s

Heap Sort Iteration 92 time taken: 0.779574 s

Heap Sort Iteration 93 time taken: 0.769772 s

Heap Sort Iteration 94 time taken: 0.711241 s

Heap Sort Iteration 95 time taken: 0.684034 s

Heap Sort Iteration 96 time taken: 0.662861 s

Heap Sort Iteration 97 time taken: 0.659606 s

Heap Sort Iteration 98 time taken: 0.565343 s

Heap Sort Iteration 99 time taken: 0.531627 s

Heap Sort Iteration 100 time taken: 0.528558 s

**Total time is taken for 100 iterations: 56.974848 s**

**The average time taken for 100 times sorting: 0.569748 seconds**