

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | | | | | | | |
|  | | Assignment-2 | | | | |  | |
|  |  | | | | | | |  |
|  | | | |  |  | | | |
|  | | | | Saad Ahmed Salim 1712299042 Section: 02 |  | | | |
|  | | | | 11-12-2020—Design & Analysis of Algorithms—Dr. Sifat Momen |  | | | |
|  | | |  | | |  | | |

**Heapsort Algorithm**

Procedure Min-Heapify(H, small)

Input: H: an array of elements; **small**: an index array.

Output: H modified such that element **small** roots a min-heap

1. left = 2\* **small** + 1
2. right = 2\***small** + 2
3. smallest = **small**
4. if left < n and arr[left] < arr[**small**] then:
5. smallest = left
6. if right < n and arr[right] < arr[smallest] then:
7. smallest = right
8. if smallest != **small** then:
9. arr[**small**], arr[smallest] = arr[smallest], arr[**small**]
10. heapify(arr, n, smallest)

Converting an array to a Min-Heap:

Build-Min-Heap(A)

1. heap-size[A] = length[A];
2. for i<-floor(length[A]/2) down to 1:
3. do Min-Heapify(A, i)

HeapSort(arr):

Build-Min-heap(Arr)

1. n <- length(arr)
2. for i<-n down to 1:
3. swap arr[i] and arr[0]
4. Min-Heapify(n, i)

**Heap Sort Code in Python**

def heapify(arr, sizeofheap, small):

smallest = small # Initialize smalls as root

left\_child = 2 \* small + 1

right\_child = 2 \* small + 2

# If left child is smaller than root

if left\_child < sizeofheap and arr[left\_child] < arr[smallest]:

smallest = left\_child

# If right child is smaller than

if right\_child < sizeofheap and arr[right\_child] < arr[smallest]:

smallest = right\_child

# If smallest is not root

if smallest != small:

(arr[small], arr[smallest]) = (arr[smallest],arr[small])

heapify(arr, sizeofheap, smallest)

def heapSort(arr, sizeofheap):

for i in range(int(sizeofheap / 2) - 1, -1, -1):

heapify(arr, sizeofheap, i)

for i in range(sizeofheap-1, -1, -1):

# Move current root to end #

arr[0], arr[i] = arr[i], arr[0]

heapify(arr, i, 0)

def printArray(arr, n):

for i in range(n):

print(arr[i], end = " ")

print()

if \_\_name\_\_ == '\_\_main\_\_':

arr = [1, 4, 6, -6, 9, 3, 0, 8]

n = len(arr)

print("Initial Array is : ")

printArray(arr,n)

heapSort(arr, n)

print("SORTED ARRAY is : ")

printArray(arr, n)