

MAX-HEAPIFY

(project two part I)

Exercises 6.2.5

The Code of MAX-HEAPIFY is quite efficient in terms of constant factors, except possibly for the recursive call in line 10, which might cause some compilers to produce inefficient code. Write an efficient MAX-HEAPIFY that uses an iterative control construct (a loop) instead of recursion.

MAX-HEAPIFY(A,i)

1. $l = \text{LEFT}(i)$
2. $r = \text{RIGHT}(i)$
3. **if** $l \leq A.\text{HEAP_size}$ and $A[l] > A[i]$
4. $\text{largest} = l$
5. **else** $\text{largest} = i$
6. **if** $r \leq A.\text{HEAP_size}$ and $A[r] > A[\text{largest}]$
7. $\text{largest} = r$
8. **if** $\text{largest} \neq i$
9. exchange $A[i]$ with $A[\text{largest}]$
10. MAX-HEAPIFY(A,largest)

Input data:

```
10          ;array size
16          ;A[0]
4           ;A[1]
10          ;....
14
7
9
3
2
8
1           ;A[9]
```

Output:

```
16
```

14
10
8
7
9
3
2
4
1

Restrictions:

- The project will be personal, do not copy or modify by the others include the one you find on internet, or you will get nothing of this project.
- Please use c/c++ only.

Deadline:

- due time: 3/31 13:00
- upload to the ftp
- Host: 140.113.215.178
- port: 1234
- username/pw: part1//part1
- let your file name be your student ID, like "9812345.cpp"

Demo time:

- 3/31 16.30 ~ 18:30 at EC324
- We will also demo the project one on the same time. Please be sure you still remember what you did in project one.

Please let me know if you have any problem or I made any mistakes.

e-mail: fd3srxs.cs99g@nctu.edu.tw

LAB: EC637

TA Joseph