ECE608, Fall 2015, Quiz 5

Last Name: ___________________ First Name: ____________________

I certify that I have neither given nor received unauthorized aid on this quiz.

Signed: ___________________

Use only the space provided on this page to answer the following question(s).
Do not write your answers on the other side of the page.

(1) The algorithms Heapify(A, i) and Build−Heap(A) that we have seen in class are given below. Modify the algorithms such that, after Build−Heap(A) is run, A[1] will be the smallest element in A. Mark your modifications clearly on the algorithms given below. Make the smallest number of modifications.

Heapify(A, i)
1  l ← Left(i)
2  r ← Right(i)
3  if l ≤ heap−size[A] and A[l] > A[i] then largest ← l
4  else largest ← i
5  if r ≤ heap−size[A] and A[r] > A[largest] then largest ← r
6  if largest ≠ i then exchange A[i] <−> A[largest]
7  Heapify(A, largest)

Build−Heap(A)
1  heap−size[A] ← length[A]
2  for i = ⌊length[A]/2⌋ downto 1 do
3  Heapify(A, i)

(2) Show the result of applying the modified Build−Heap(A), with the modified Heapify(A, i), to the following instance. Provide your answer in the form of a binary tree.
ECE608, Fall 2015, Quiz 5

Last Name: ___________________ First Name: ____________________

I certify that I have neither given nor received unauthorized aid on this quiz.

Signed: ___________________

Use only the space provided on this page to answer the following question(s).
Do not write your answers on the other side of the page.

(1) The algorithms Heapify(A, i) and Build − Heap(A) that we have seen in class are given below. Modify the algorithms such that, after Build − Heap(A) is run, A[1] will be the smallest element in A. Mark your modifications clearly on the algorithms given below. Make the smallest number of modifications.

Heapify(A, i)
1. \( l \leftarrow \text{Left}(i) \)
2. \( r \leftarrow \text{Right}(i) \)
4. else \( \text{largest} \leftarrow i \)
5. if \( r \leq \text{heap−size}[A] \) and \( A[r] > A[\text{largest}] \) then \( \text{largest} \leftarrow r \) \( A[r] < A[\text{largest}] \)
6. if \( \text{largest} \neq i \) then exchange \( A[i] < \rightarrow A[\text{largest}] \)
7. Heapify(A, \text{largest})

Build − Heap(A)
1. \( \text{heap−size}[A] \leftarrow \text{length}[A] \)
2. for \( i = \lceil \text{length}[A]/2 \rceil \) downto 1 do
3. Heapify(A, i)

(2) Show the result of applying the modified Build − Heap(A), with the modified Heapify(A, i), to the following instance. Provide your answer in the form of a binary tree.