

42

41

23

17

6



42

33

37

18

17

4

*H1* and *H2* are max heaps. A max heap is a priority queue where the biggest key is in the root, and the definition of the heap order is that any node must have a key that is bigger than all its descendants’.

Draw *H1* after a *deleteMax* operation.

6

9

41



42

41

23

17

6

Draw *H2* after an *insert* operation with key 35.

18

33

35



42

33

37

18

17

4

3

# Exercise 2

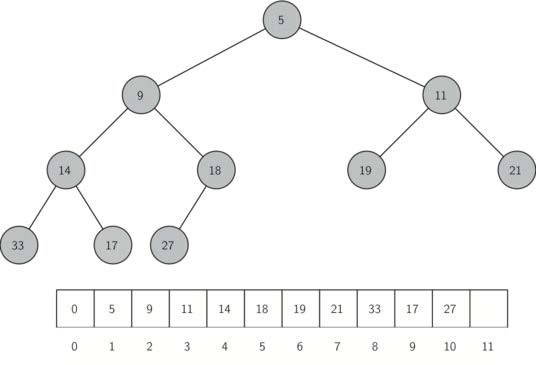
A priority queue can be implemented as a simple array of integers in which the first element is left unused for addressing purposes. Can the array below represent a priority queue? Please explain your answer.

{0,17,21,23,44,32,65,38,56,46,69,33,77,67,56,39,61,60,62,50,71}

No, this array cannot represent a max-heap priority queue because the root (17) is smaller than its children.

# Exercise 3

The figure below represents a priority queue implemented in a simple array.



Show the contents of the priority queue (draw it or show the array) after the following three operations have been performed: first you must add an element with the value of 7 (insert(7)), then an element with the value of 15 is added (insert(15)), and finally the smallest element is removed (deleteMin()).

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 5 | 9 | 11 | 14 | 18 | 19 | 21 | 33 | 17 | 27 |  |  |  |
| 0 | 5 | 7 | 11 | 14 | 9 | 19 | 21 | 33 | 17 | 27 | 18 |  |  |
| 0 | 5 | 7 | 11 | 14 | 9 | 15 | 21 | 33 | 17 | 27 | 18 | 19 |  |
| 0 | 7 | 9 | 11 | 14 | 18 | 15 | 21 | 33 | 17 | 27 | 19 |  |  |



32

10

6



29

10

6

*H1* and *H2* are max heaps. A max heap is a priority queue where the biggest key is in the root, and the definition of the heap order is that any node must have a key that is bigger than all its descendants’.

Draw *H1* after an *insert* operation with key 12.

10

12



32

10

6

1

Draw *H2* after a *deleteMax* operation.



29

10

6

3

8

6

9

10



29

10

6

# Exercise 5

Consider the following array:

{0,4,17,12,20,25,15,38,22,30,24,45,67,18,40,42,36,56}

Can this array represent a *heap* (a priority queue)? Explain your answer.

* **Not a max-heap:** the root at index 1 is 4; its children at indices 2 and 3 are 17 and 12, both larger than 4
* **Not a min-heap:** check index 5 (value 25). Its left child is at index 10 with value 24

Because it violates both heap orders, the array cannot represent a valid heap (priority queue).