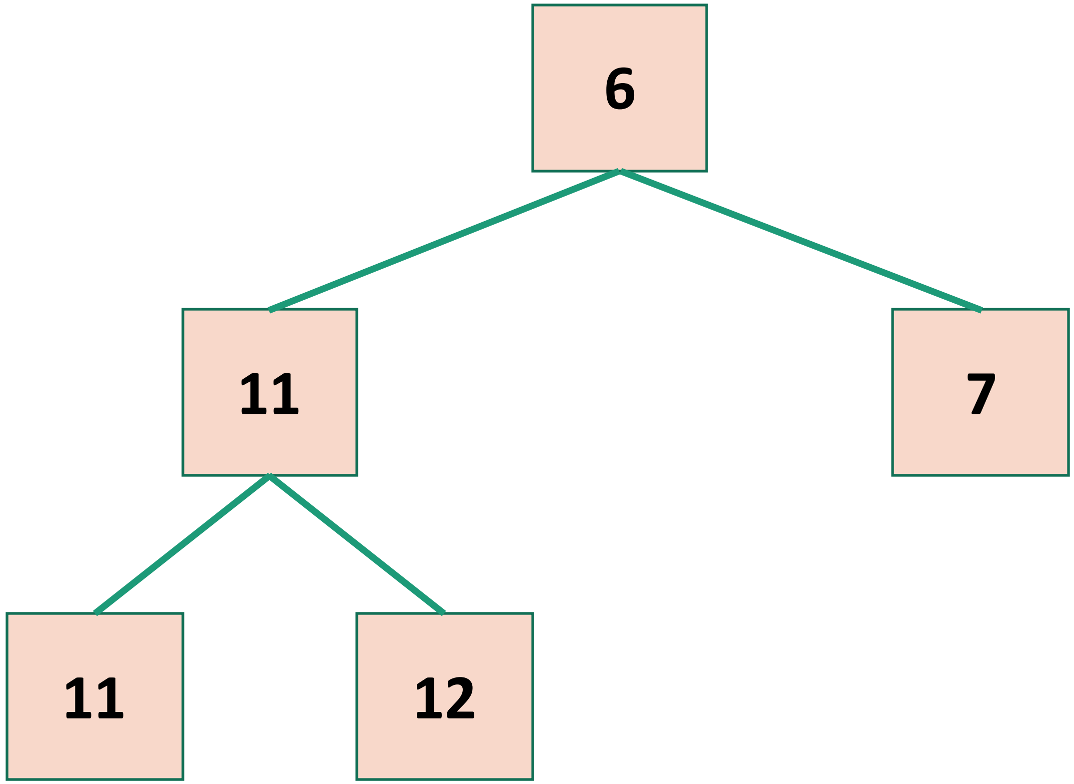
Priority Queue and Heap

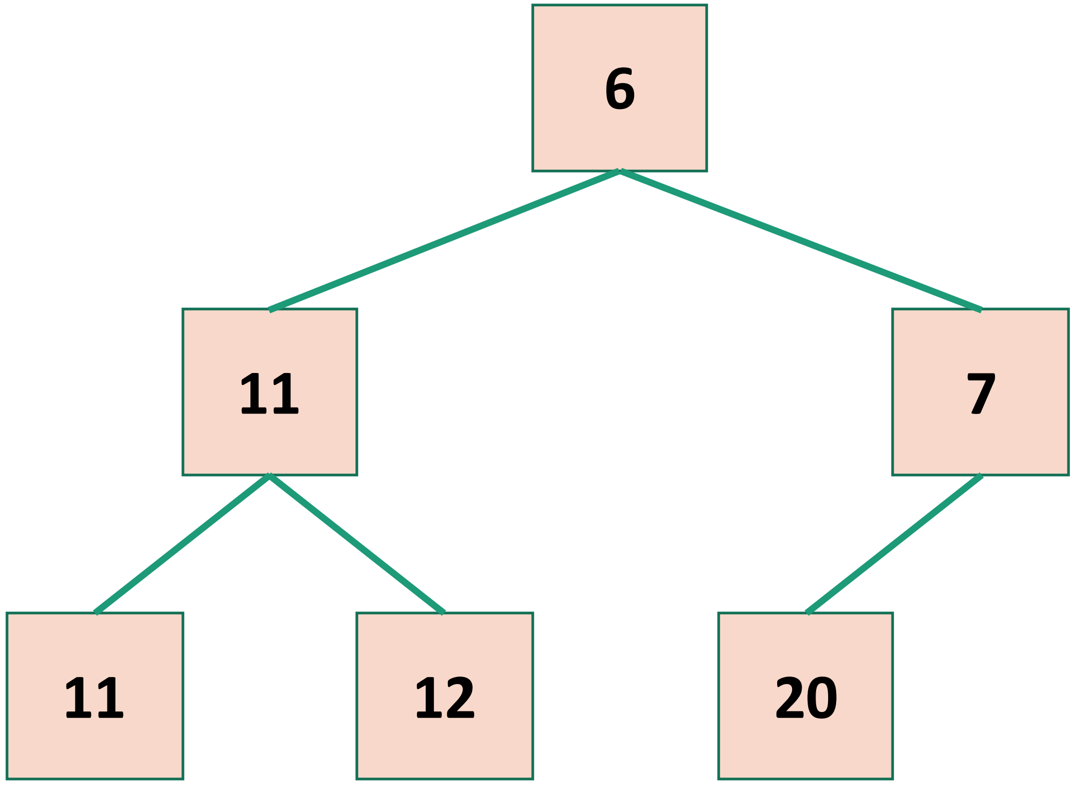
Some slides were created by Dr. Jianfeng Ren.

Edited by Heshan Du

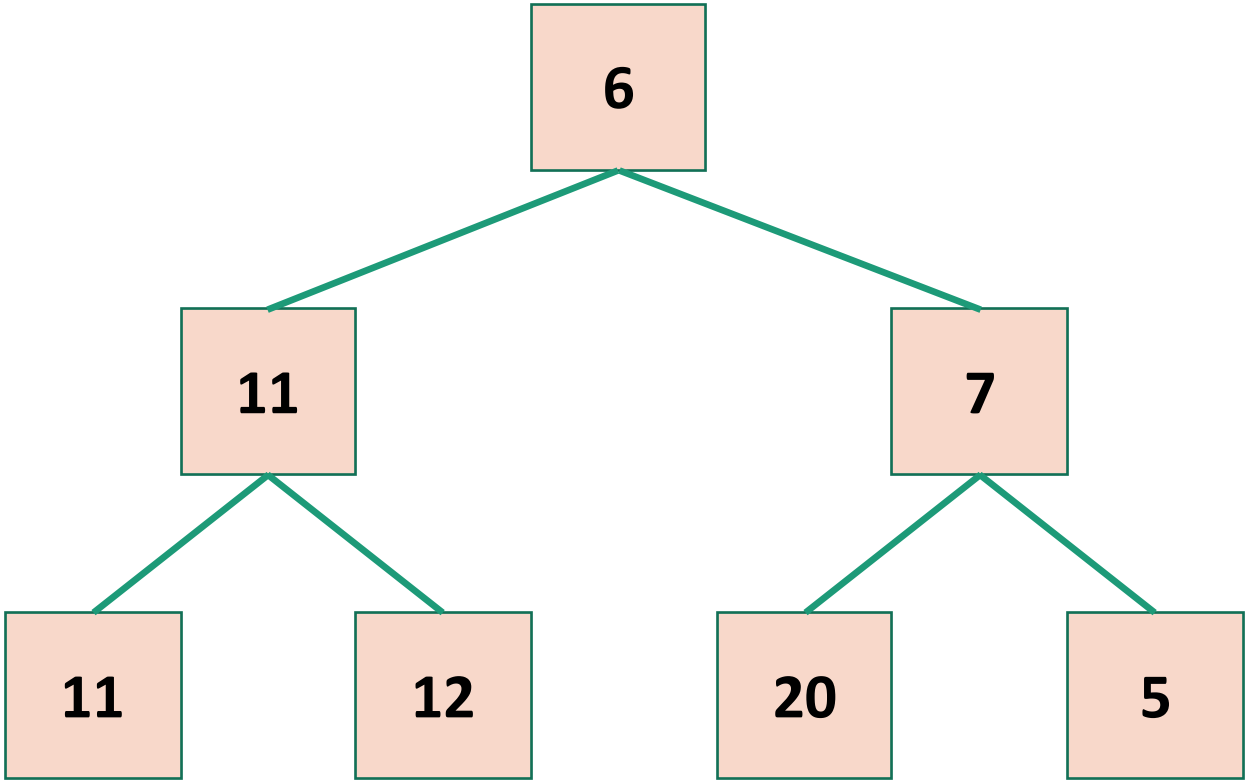
Exercise 1: insert 20, 5, 4



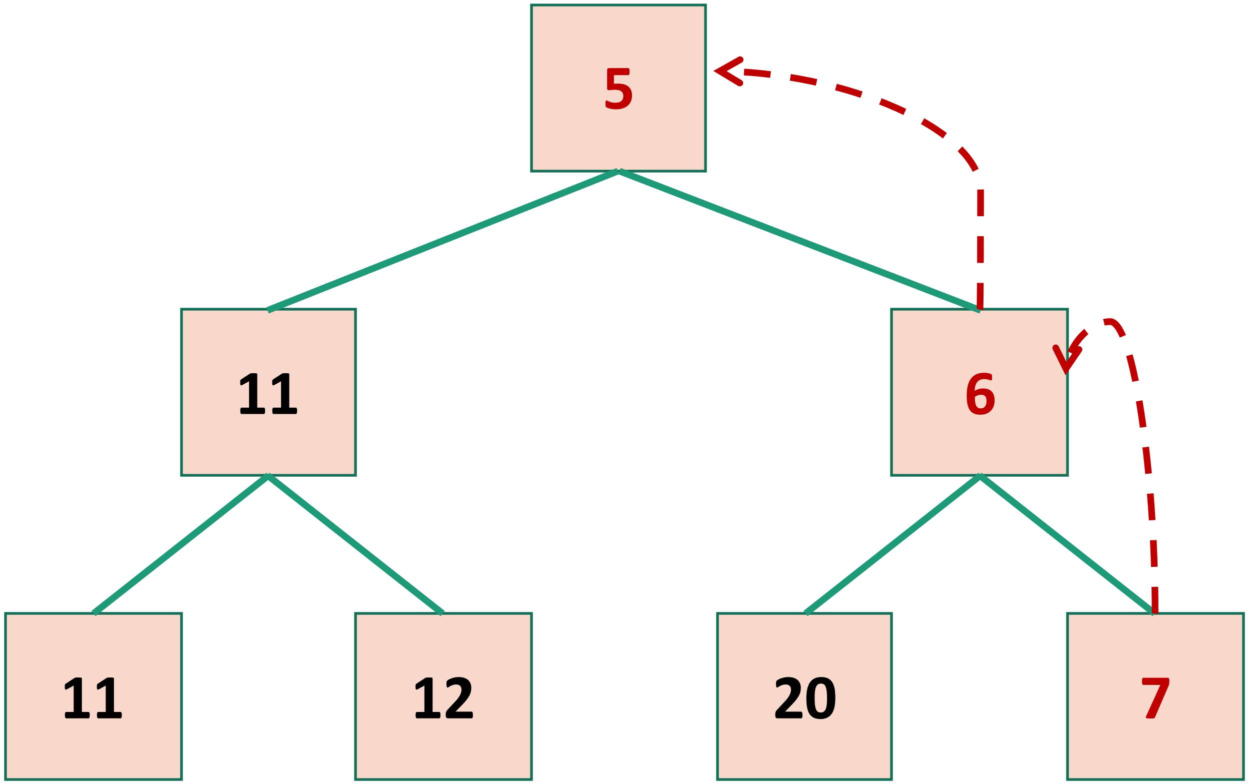
insert 20



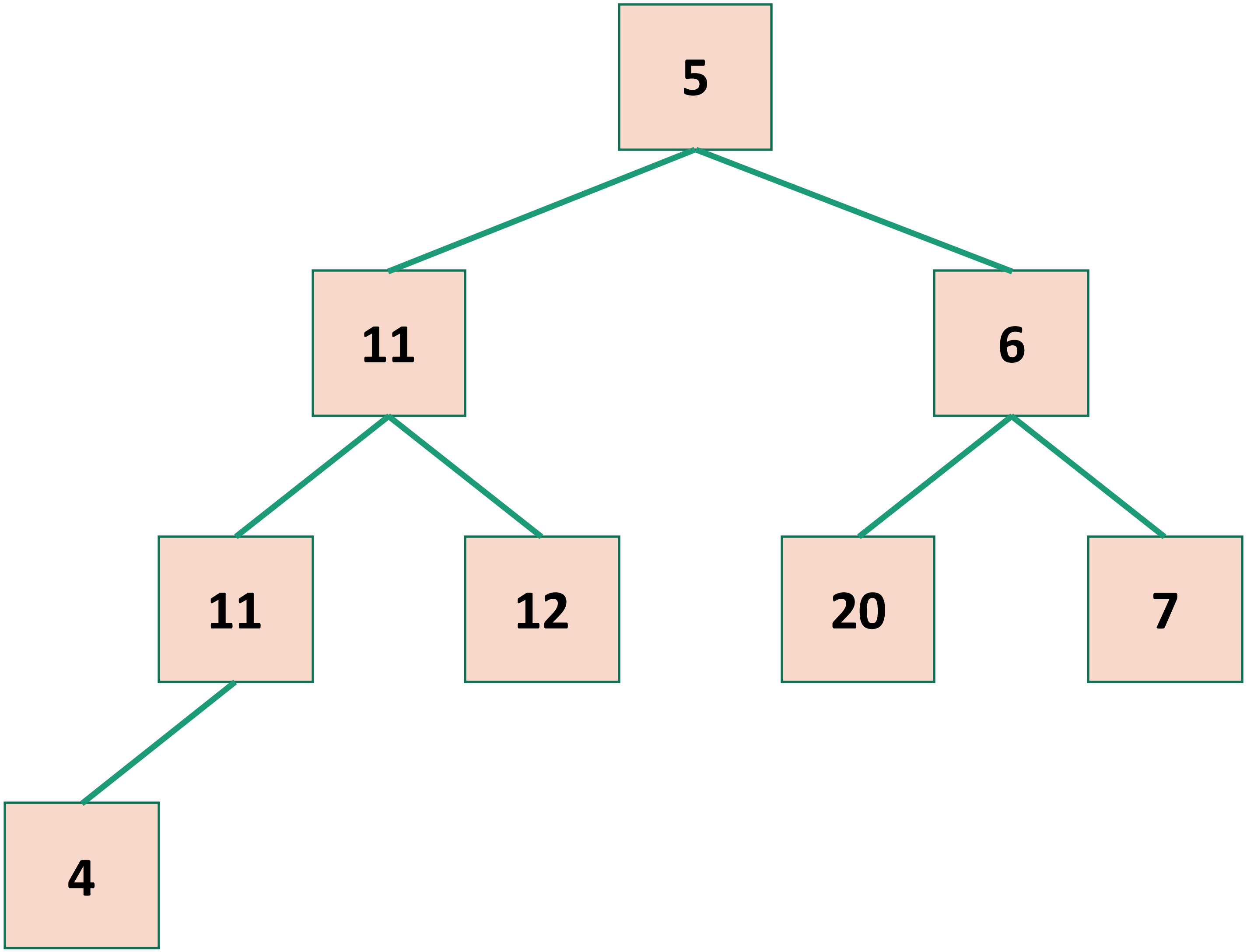
insert 5



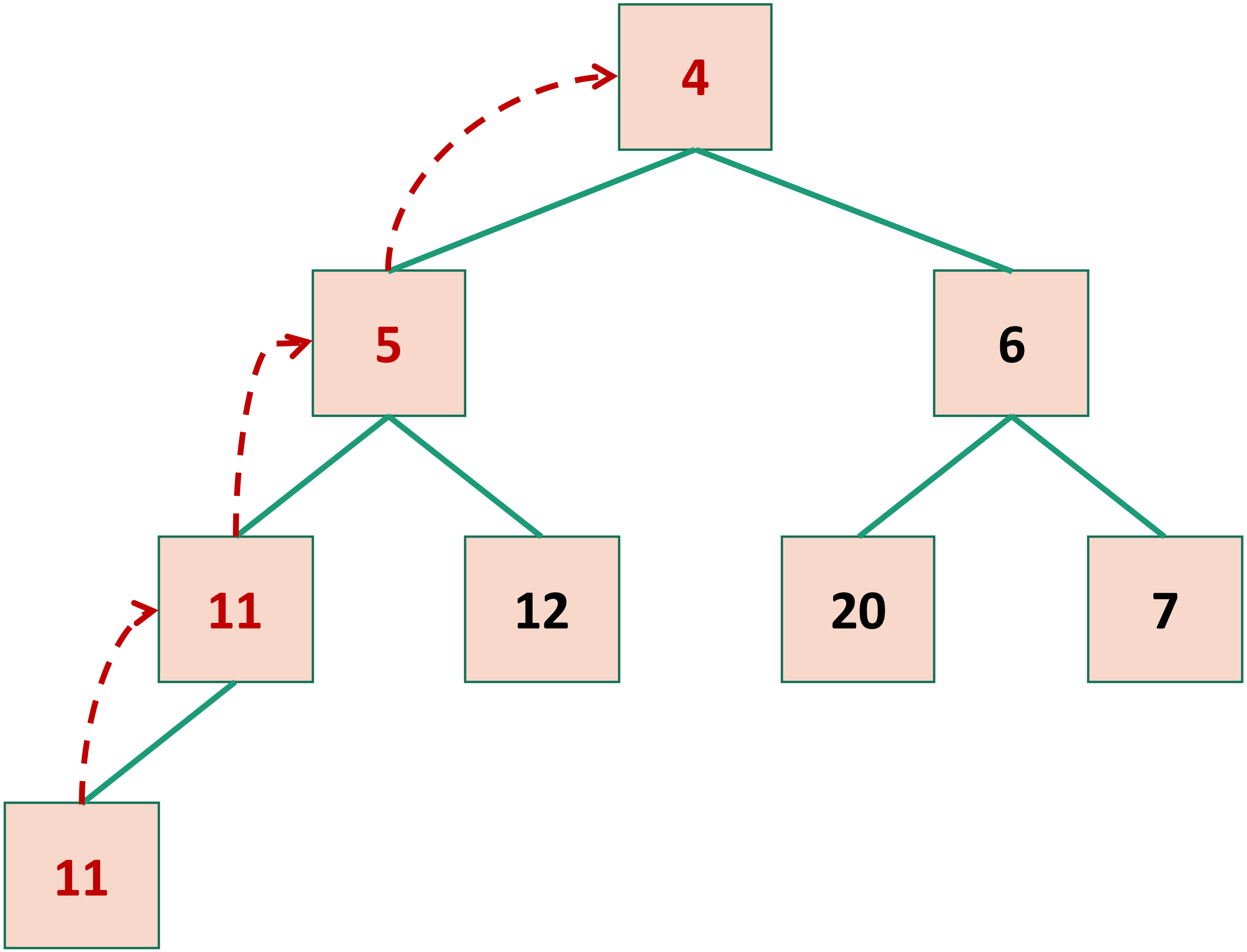
insert 5



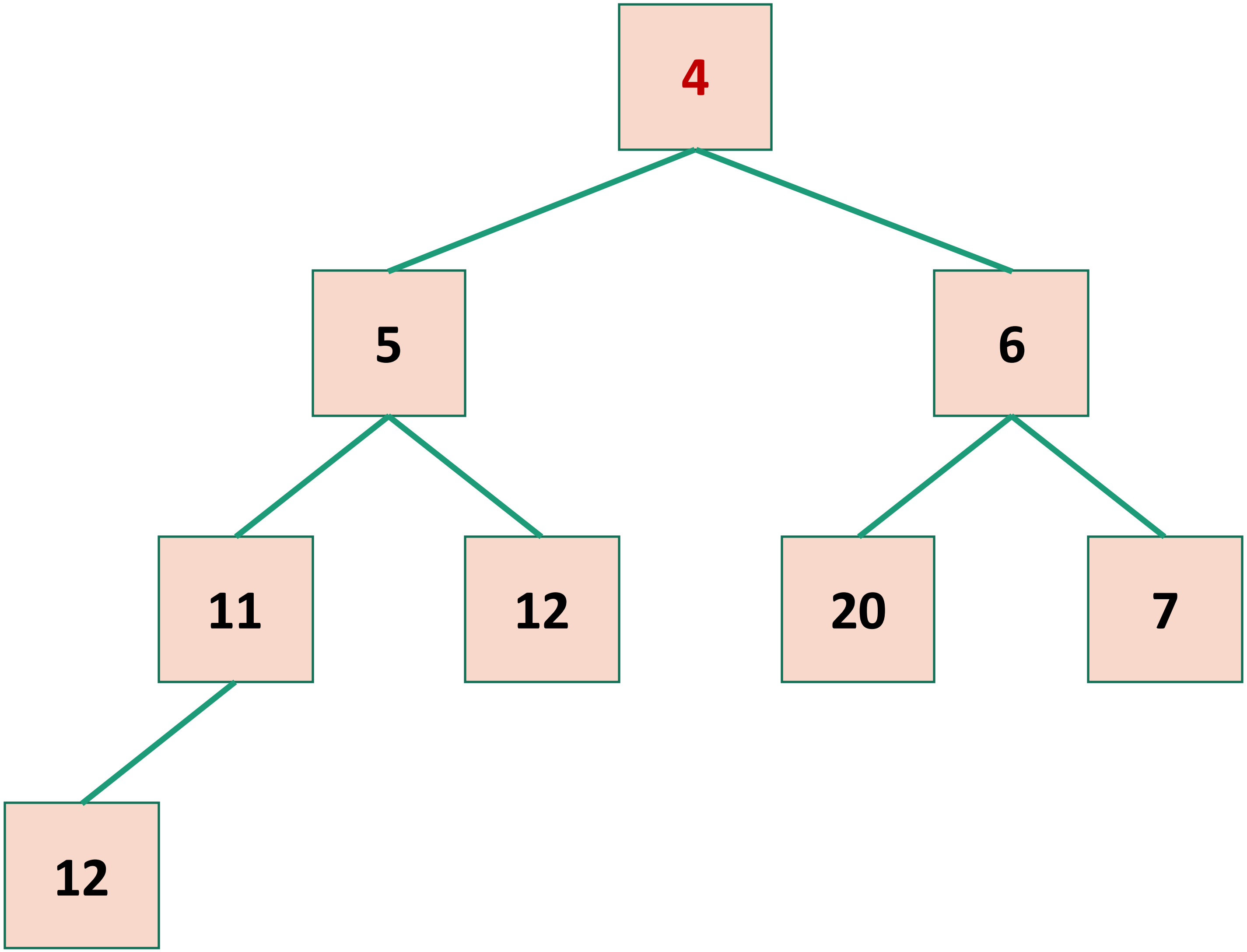
insert 4



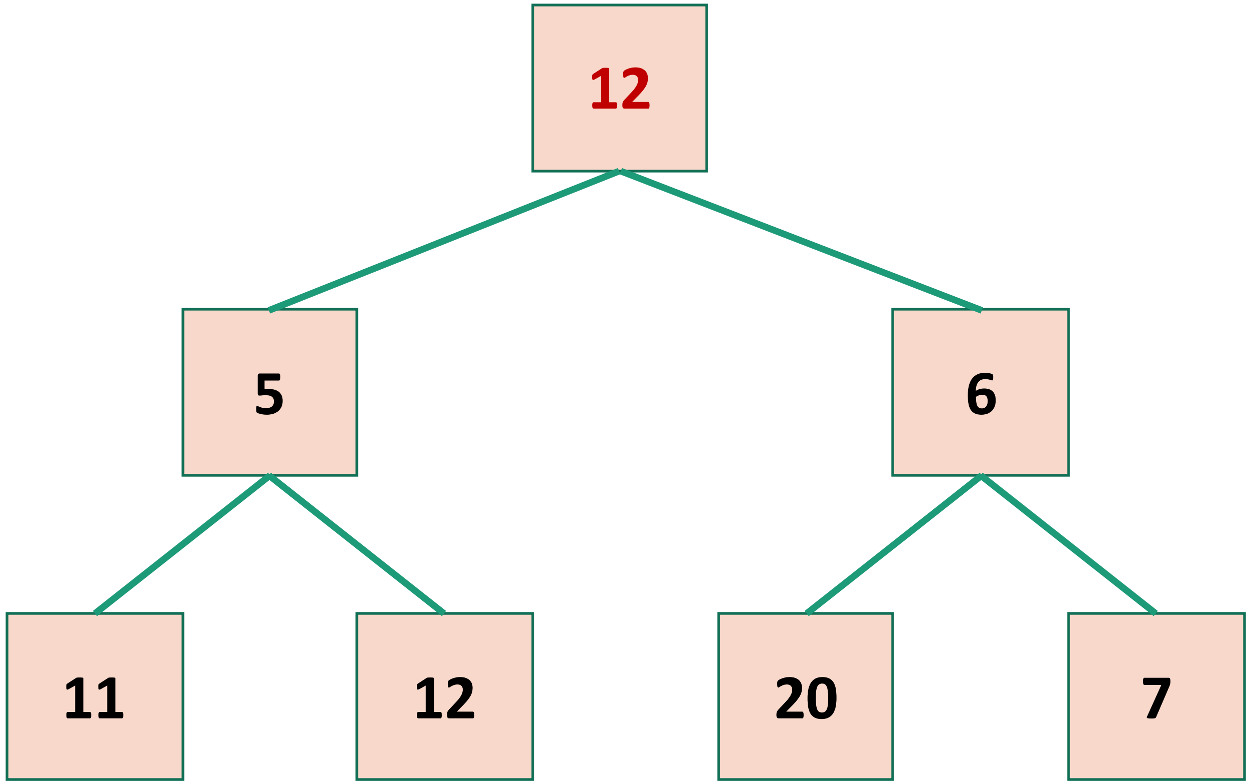
insert 4



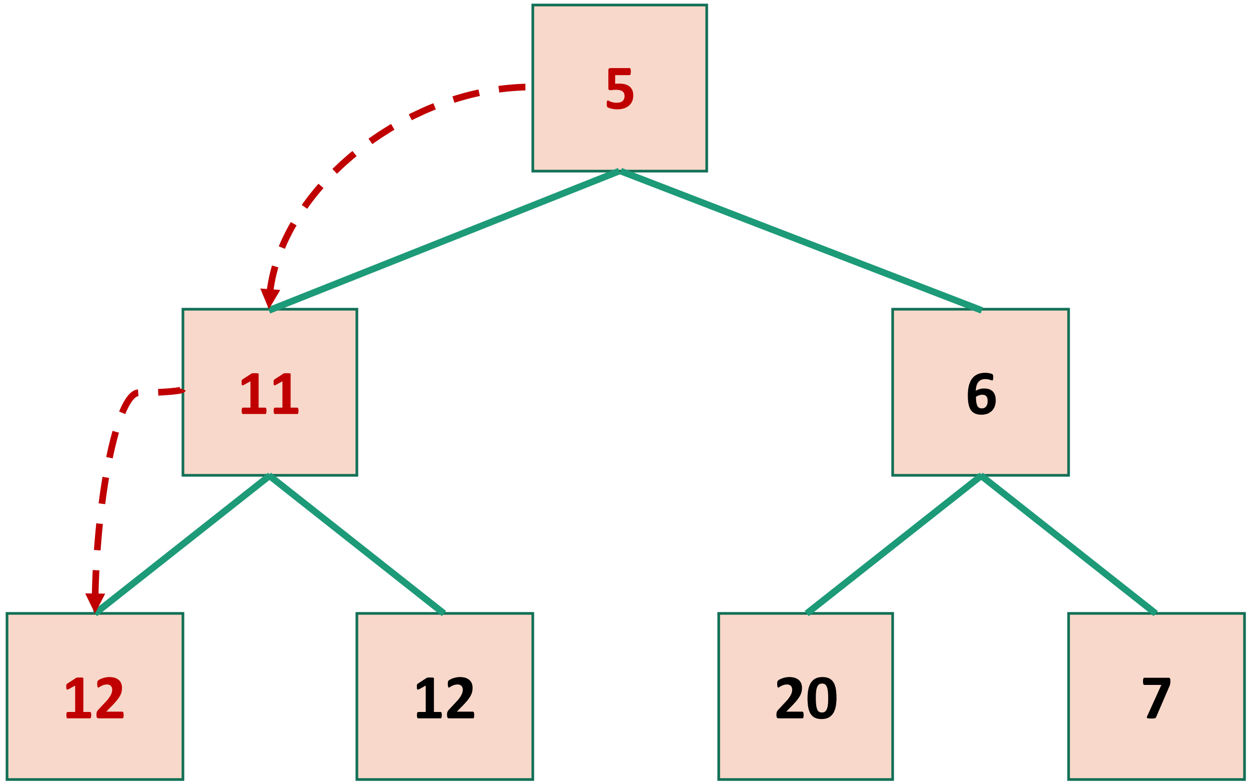
Exercise 2: remove 4



remove 4



remove 4



Exercise 3: heap sort

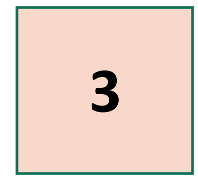
要使用 **in-place heap sort** 将序列 [3, 6, 9, 2, 5, 8] 排序为**非递增顺序（从大到小）**

- Sort the following sequence in non-increasing order using in-place heap sort:

[3 6 9 2 5 8]

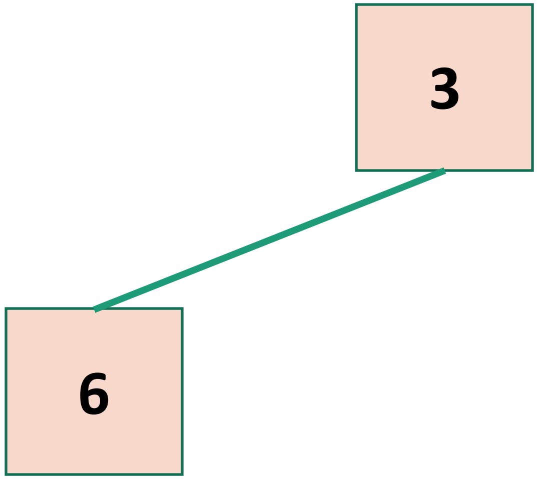
heap sort - phase 1

- Original sequence: [3 6 9 2 5 8]
- Array: [**3** 6 9 2 5 8]



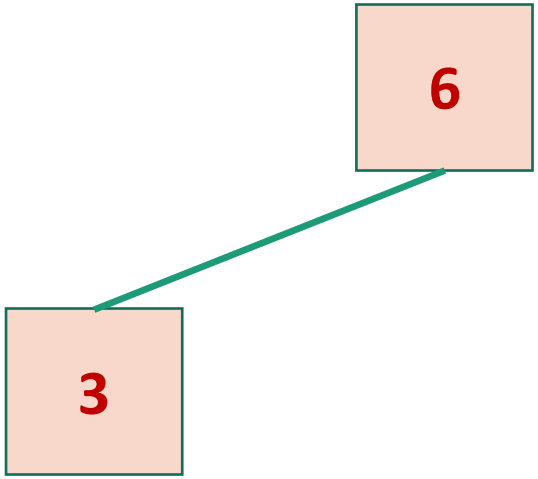
heap sort - phase 1

- Original sequence: [3 6 9 2 5 8]
- Array: [**3 6** 9 2 5 8]



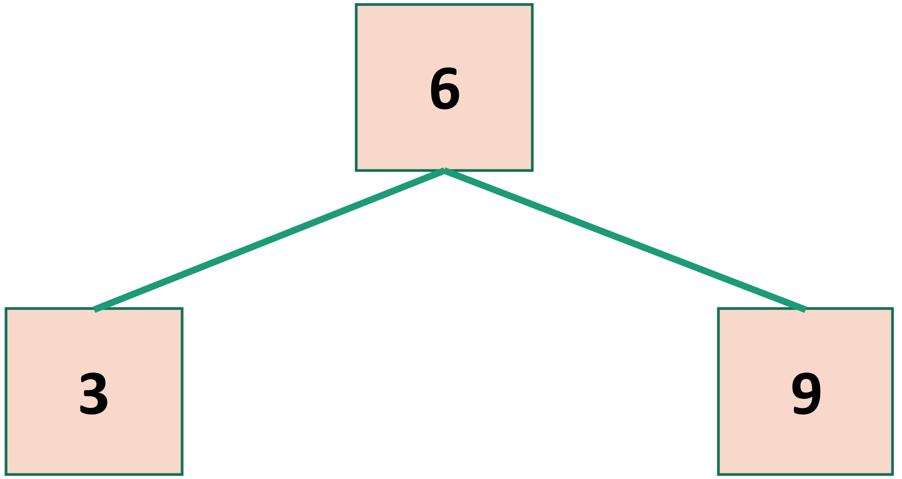
heap sort - phase 1

- Original sequence: [3 6 9 2 5 8]
- Array: [**6 3** 9 2 5 8]



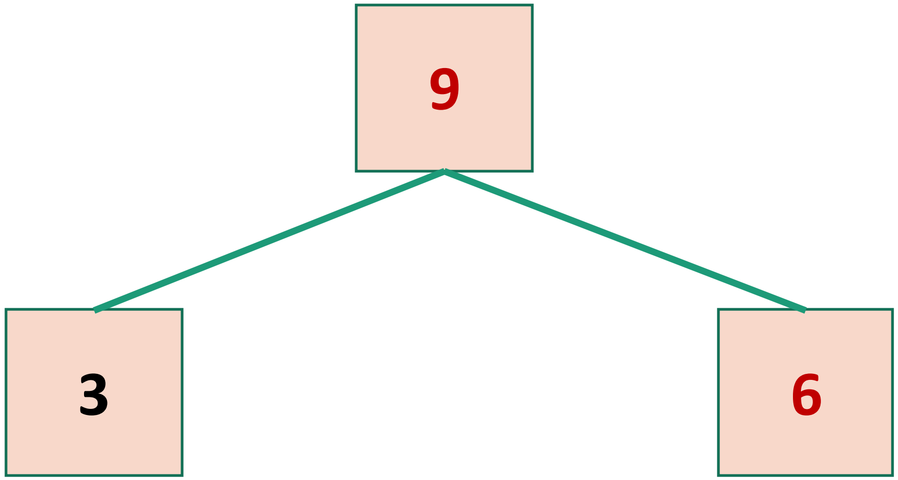
heap sort - phase 1

- Original sequence: [3 6 9 2 5 8]
- Array: [<u>**6 3 9**</u> 2 5 8]



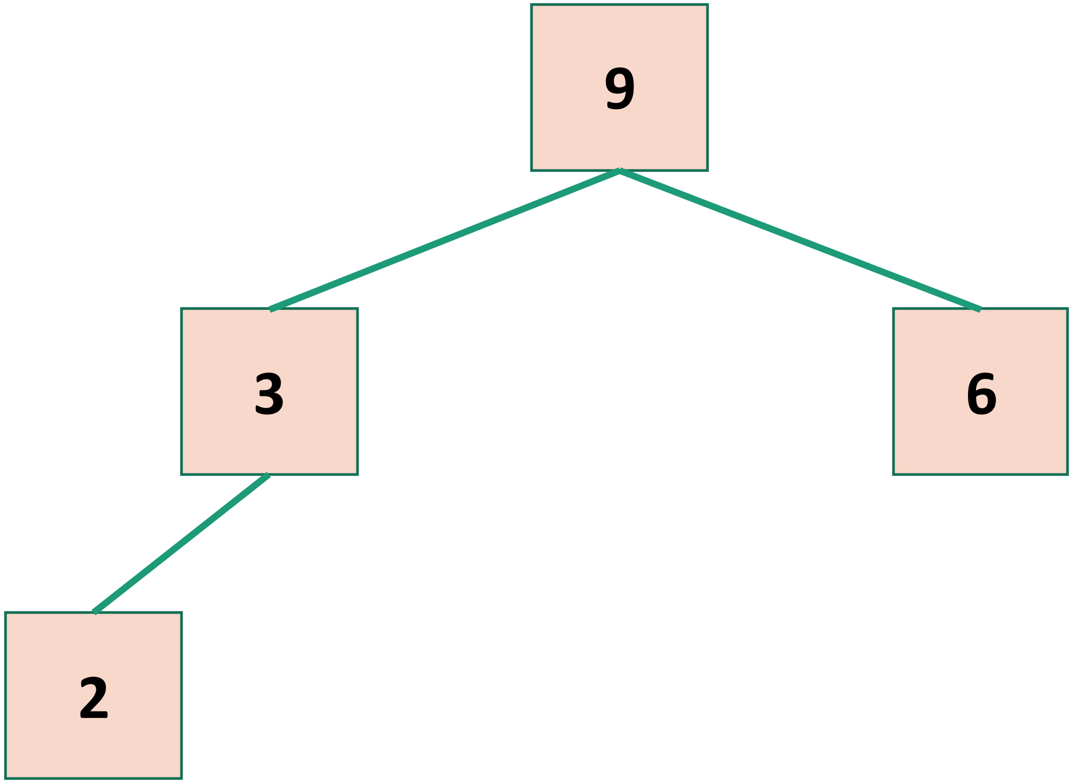
heap sort - phase 1

- Original sequence: [3 6 9 2 5 8]
- Array: [**9 3 6** 2 5 8]



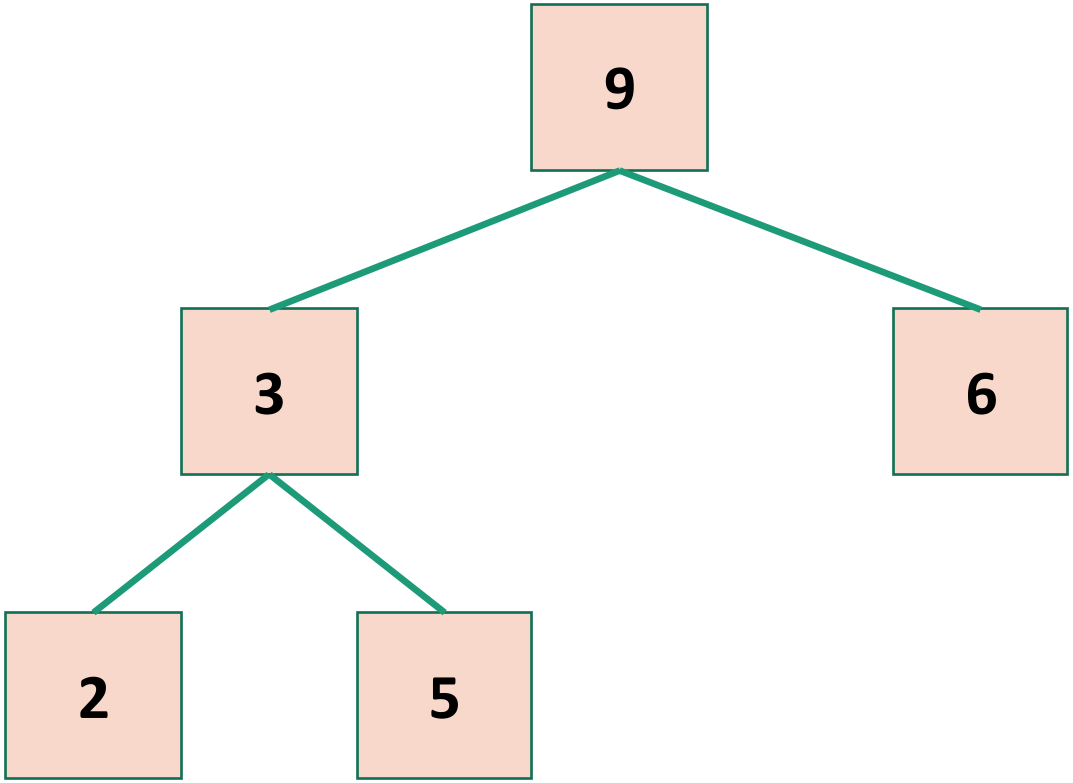
heap sort - phase 1

- Original sequence: [3 6 9 2 5 8]
- Array: [**9 3 6 2** 5 8]



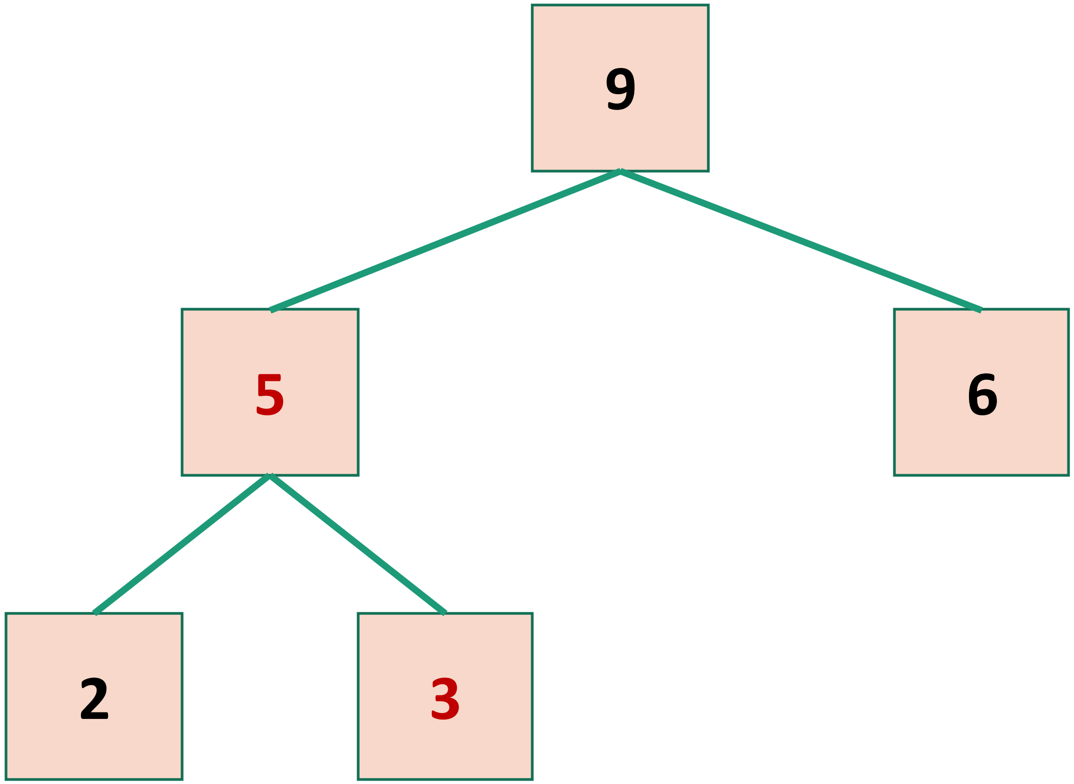
heap sort - phase 1

- Original sequence: [3 6 9 2 5 8]
- Array: [**9 3 6 2 5** 8]



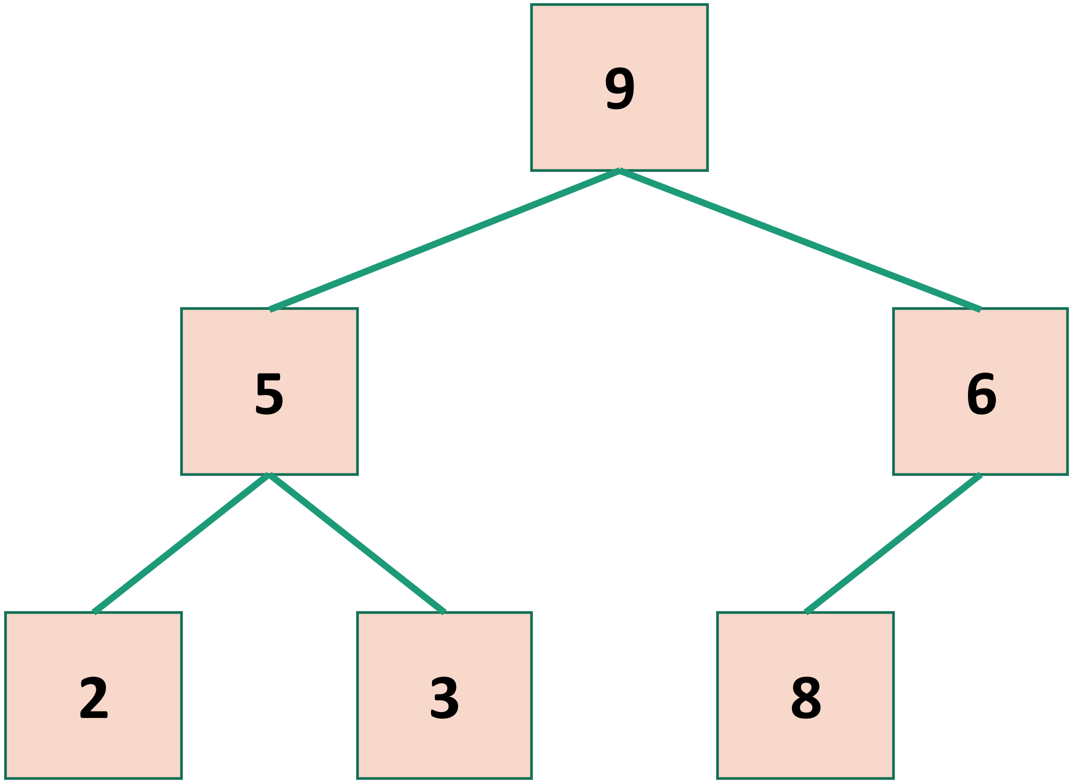
heap sort - phase 1

- Original sequence: [3 6 9 2 5 8]
- Array: [**9 5 6 2 3** 8]



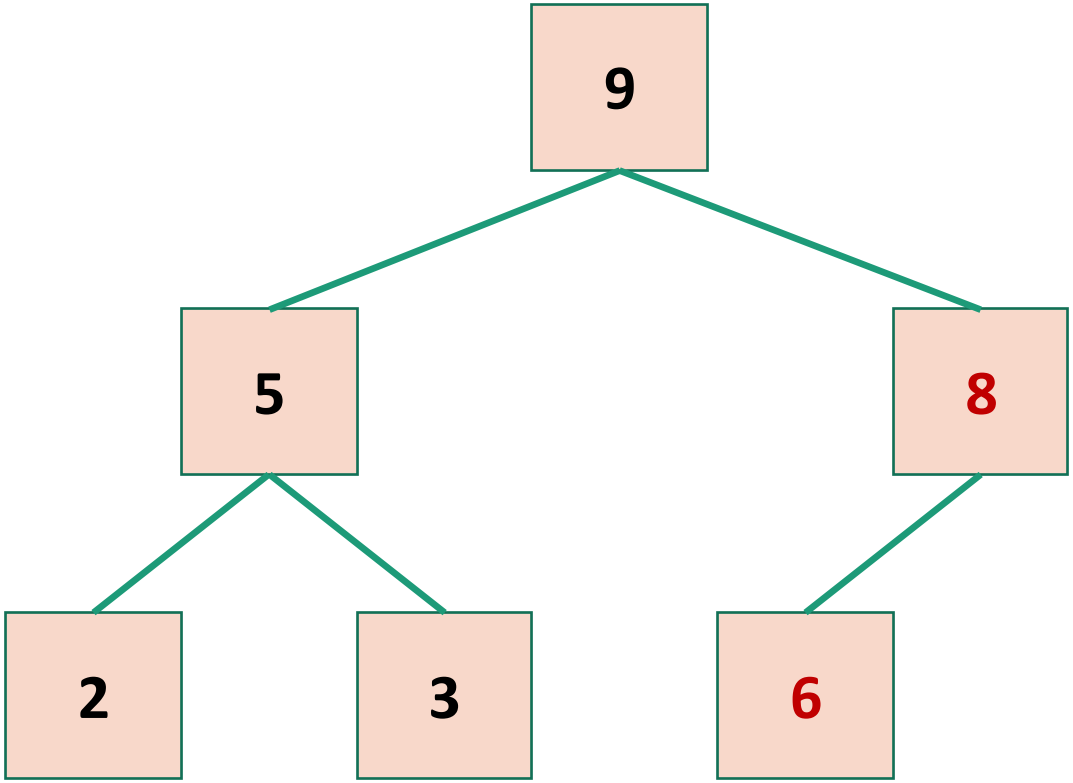
heap sort - phase 1

- Original sequence: [3 6 9 2 5 8]
- Array: [**9 5 6 2 3 8**]



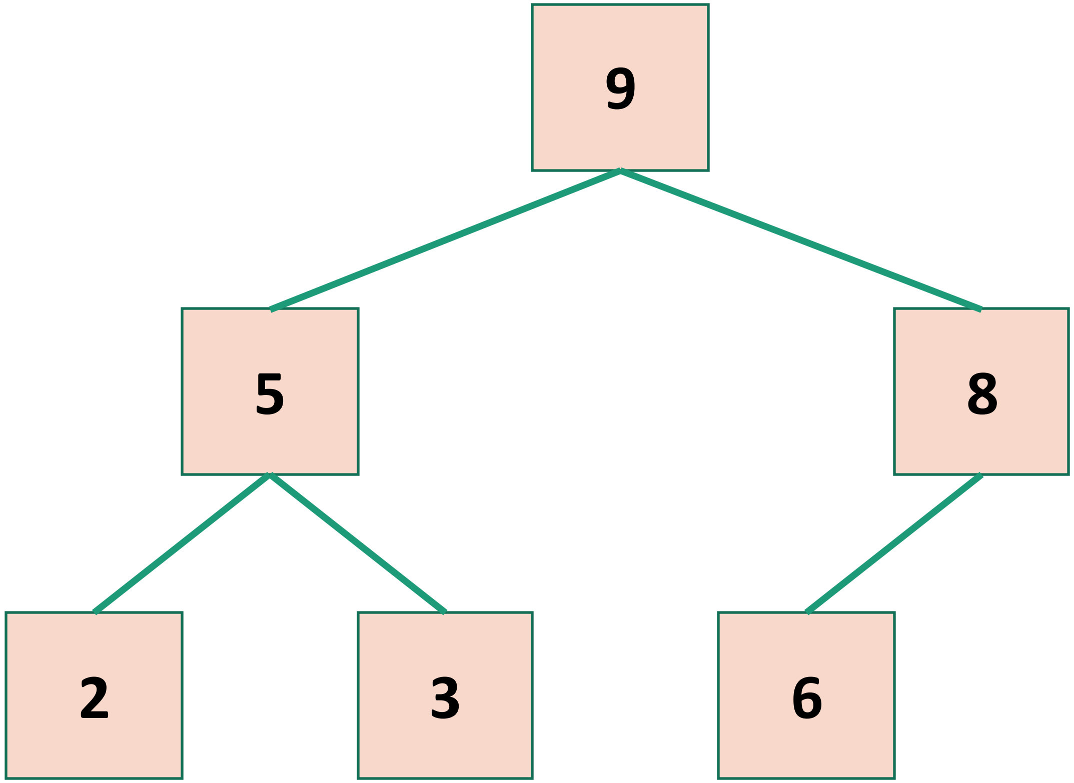
heap sort - phase 1

- Original sequence: [3 6 9 2 5 8]
- Array: [**9 5 8 2 3 6**]



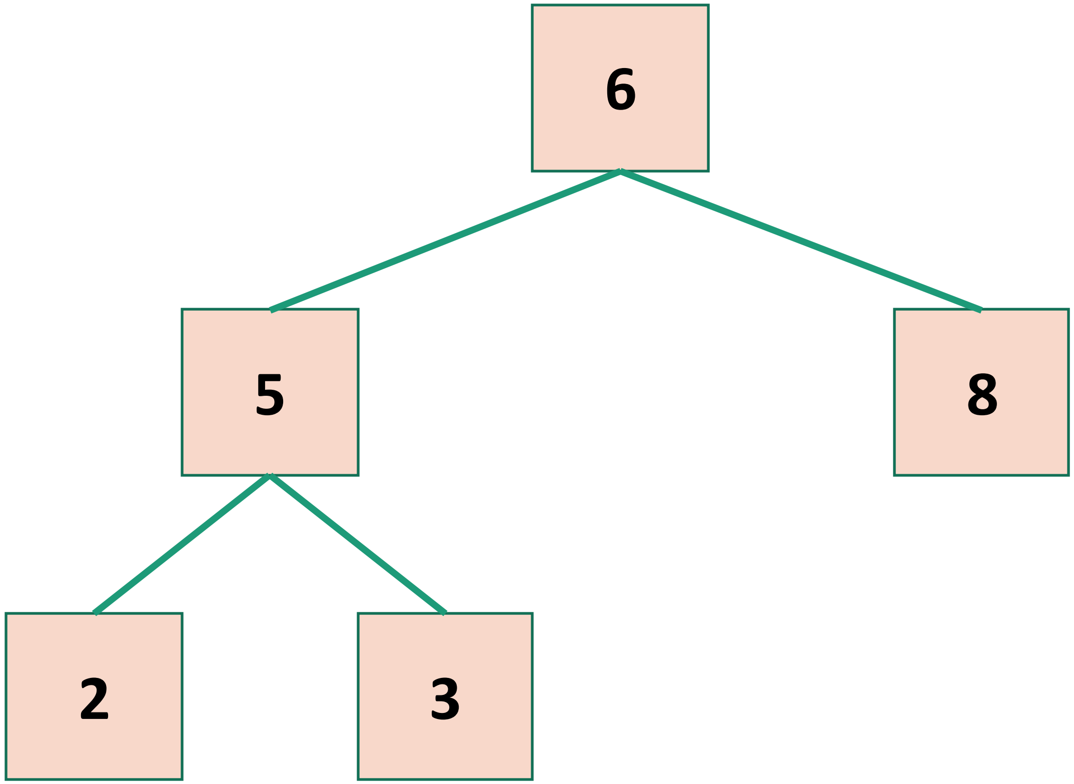
heap sort - phase 2

- Original heap: [9 5 8 2 3 6]
- Array: [9 5 8 2 3 6]



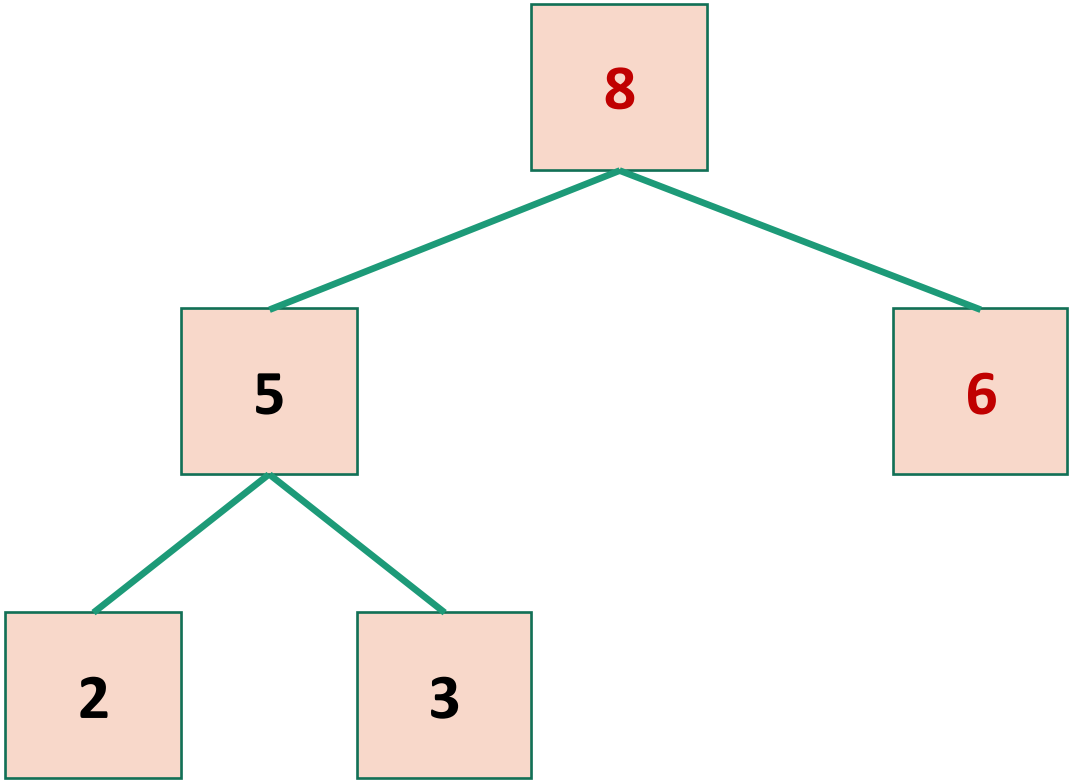
heap sort - phase 2

- Original heap: [**9 5 8 2 3 6**]
- Array: [***9*** **6 5 8 2 3**]



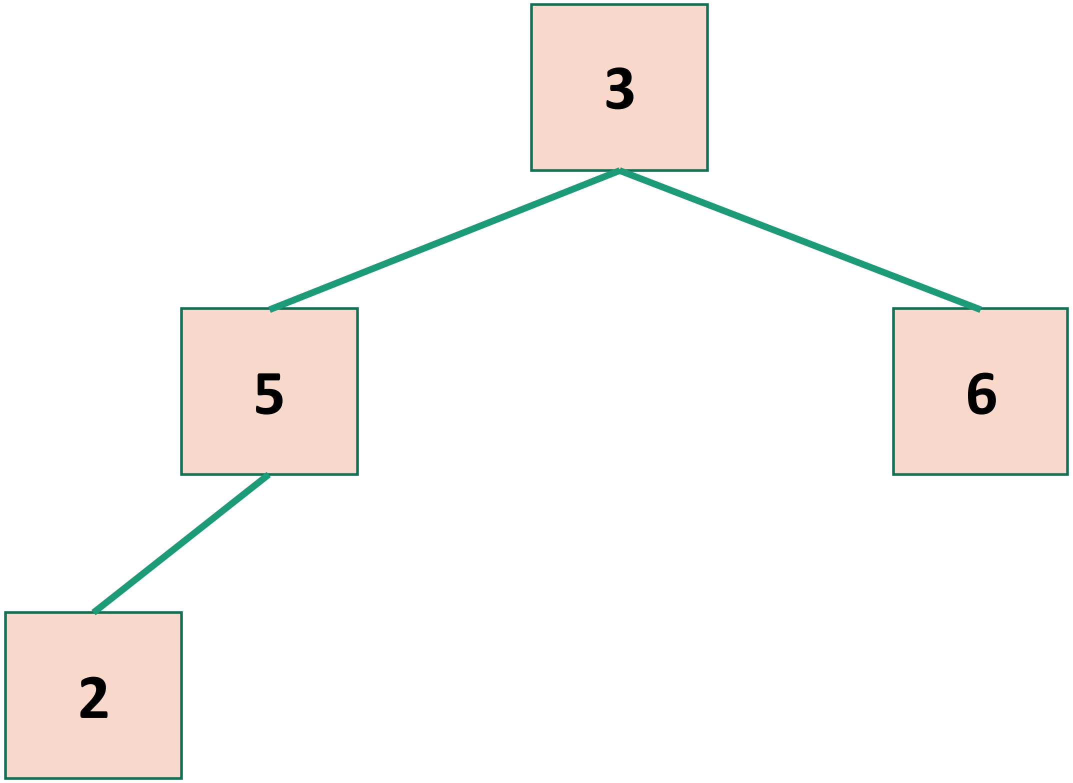
heap sort - phase 2

- Original heap: [**9 5 8 2 3 6**]
- Array: [***9*** **8 5 6 2 3**]



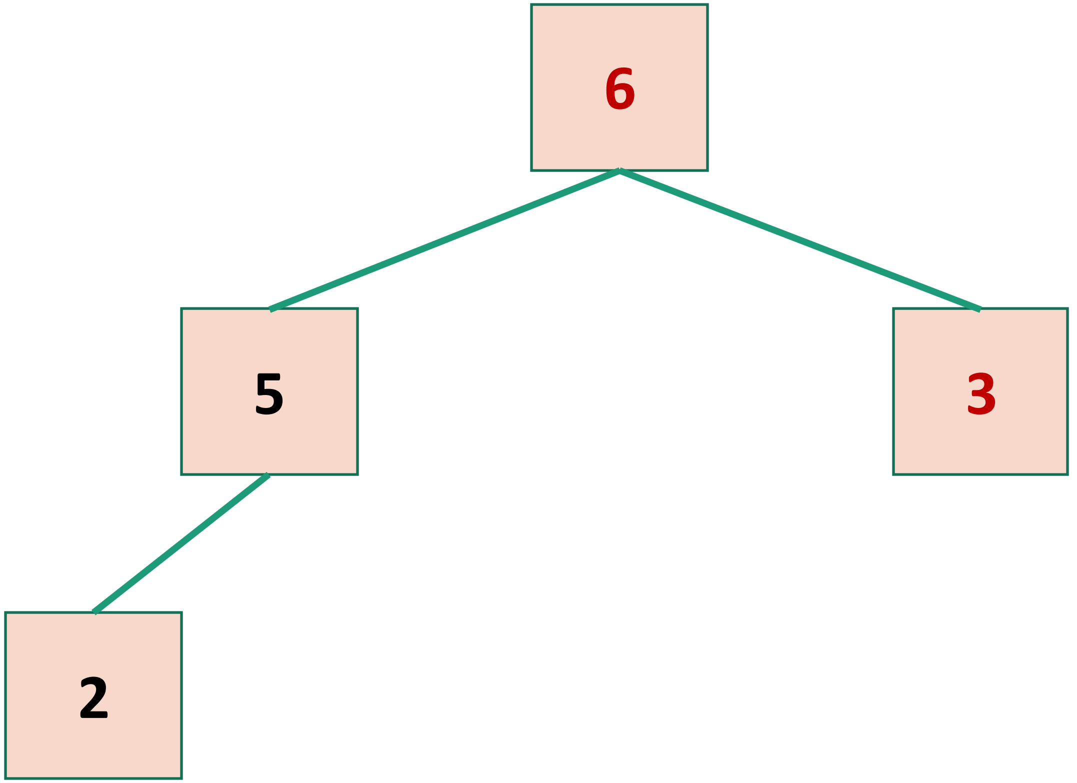
heap sort - phase 2

- Original heap: [**9 5 8 2 3 6**]
- Array: [***9 8*** **3 5 6 2**]



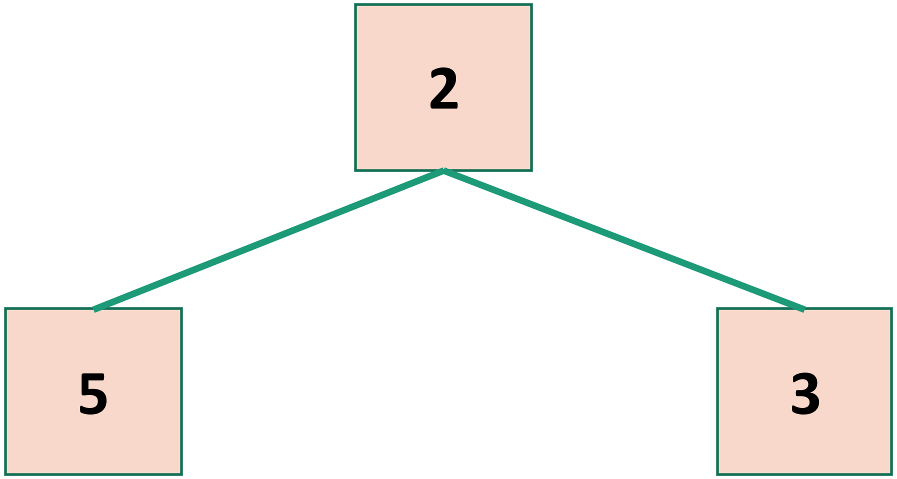
heap sort - phase 2

- Original heap: [**9 5 8 2 3 6**]
- Array: [***9 8*** **6 5 3 2**]



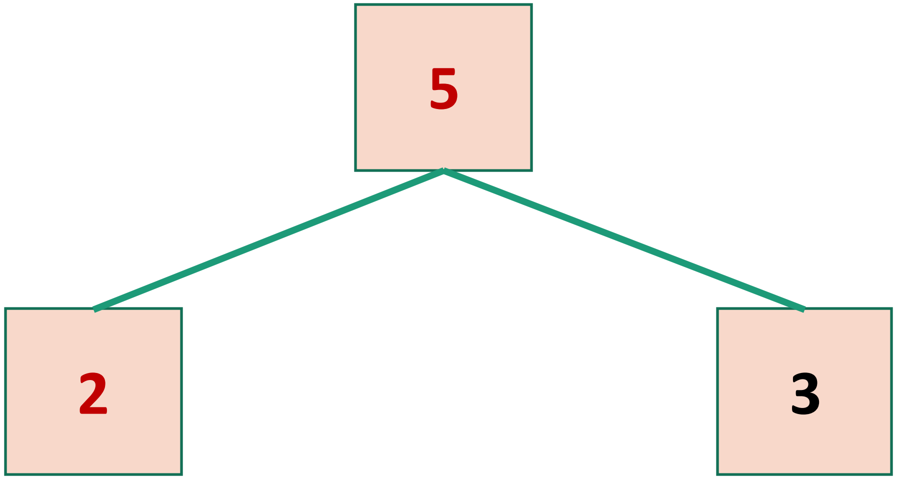
heap sort - phase 2

- Original heap: [**9 5 8 2 3 6**]
- Array: [***9 8 6*** **2 5 3**]



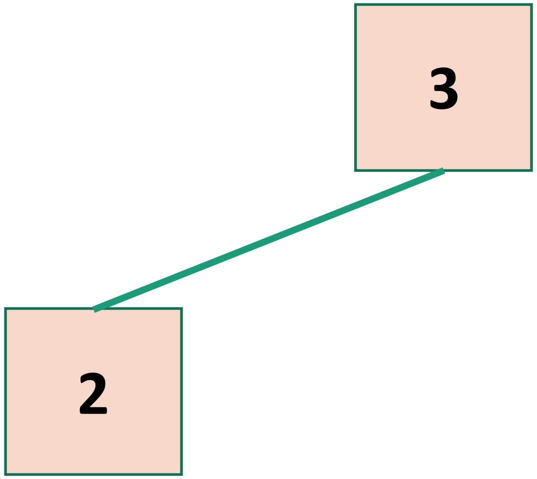
heap sort - phase 2

- Original heap: [**9 5 8 2 3 6**]
- Array: [***9 8 6*** **5 2 3**]



heap sort - phase 2

- Original heap: [**9 5 8 2 3 6**]
- Array: [***9 8 6 5*** **3 2**]



heap sort - phase 2

- Original heap: [**9 5 8 2 3 6**]
- Array: [***9 8 6 5 3*** **2**]

2

heap sort - phase 2

- Original heap: [**9 5 8 2 3 6**]
- Array: [***9 8 6 5 3 2***]

Exercise 4

Illustrate the execution of the selection-sort algorithm on the following input sequence:

(22, 15, 36, 44, 10, 3, 9, 13, 29, 25).

每轮从未排序部分中**选择最小值**放到当前起始位置。总共进行 n-1 轮比较。

Exercise 4: Solution

22	15	36	44	10	3	9	13	29	25	
3	15	36	44	10	22	9	13	29	25	
3	9	36	44	10	22	15	13	29	25	
3	9	10	44	36	22	15	13	29	25	
3	9	10	13	36	22	15	44	29	25	
3	9	10	13	15	22	36	44	29	25	
3	9	10	13	15	22	36	44	29	25	
3	9	10	13	15	22	25	44	29	36	
3	9	10	13	15	22	25	29	44	36	
3	9	10	13	15	22	25	29	36	44	

Exercise 5

Illustrate the execution of the insertion-sort algorithm on the following input sequence:

(22, 15, 36, 44, 10, 3, 9, 13, 29, 25).

从左往右依次将元素插入到左侧已经排好序的子序列中。
每一步都保持左侧序列有序。

步骤	已排序部分（左侧）	当前插入元素	操作说明
初始	—	—	[22, 15, 36, 44, 10, 3, 9, 13, 29, 25]
①	[22]	15	插入前面：→ [15, 22]
②	[15, 22]	36	保持原位：→ [15, 22, 36]
③	[15, 22, 36]	44	保持原位：→ [15, 22, 36, 44]
④	[15, 22, 36, 44]	10	插入最前面：→ [10, 15, 22, 36, 44]
⑤	[10, 15, 22, 36, 44]	3	插入最前面：→ [3, 10, 15, 22, 36, 44]
⑥	[3, 10, 15, 22, 36, 44]	9	插入 10 前面：→ [3, 9, 10, 15, 22, 36, 44]
⑦	[3, 9, 10, 15, 22, 36, 44]	13	插入 15 前面：→ [3, 9, 10, 13, 15, 22, 36, 44]
⑧	[3, 9, 10, 13, 15, 22, 36, 44]	29	插入 36 前面：→ [3, 9, 10, 13, 15, 22, 29, 36, 44]
⑨	[3, 9, 10, 13, 15, 22, 29, 36, 44]	25	插入 29 前面：→ [3, 9, 10, 13, 15, 22, 25, 29, 36, 44]

Exercise 5: Solution

22	15	36	44	10	3	9	13	29	25
15	22	36	44	10	3	9	13	29	25
15	22	36	44	10	3	9	13	29	25
10	15	22	36	44	3	9	13	29	25
3	10	15	22	36	44	9	13	29	25
3	9	10	15	22	36	44	13	29	25
3	9	10	13	15	22	36	44	29	25
3	9	10	13	15	22	29	36	44	25
3	9	10	13	15	22	25	29	36	44

Exercise 6

Show that the sum $\sum_{i=1}^{n} \log i$, appearing in the analysis of heap-sort, is $\mathrm{O}(n \log n)$.

$$\sum_{i=1}^{n} \log i \approx \int_{1}^{n} \log x \, dx$$

计算积分：

$$\int_{1}^{n} \log x \, dx = x \log x - x \,|_{1}^{n} = n \log n - n + 1$$

因此：

$$\sum_{i=1}^{n} \log i = \Theta(n \log n)$$

从而我们有：

$$\sum_{i=1}^{n} \log i = O(n \log n)$$

Exercise 6: Solution

Proof: To show $\sum_{i=1}^{n} \log i$ is $\mathrm{O}(n \log n)$, by the definition of O, we need to show that there exist a positive real constant c and a positive integer n_0 such that for all $n \geq n_0$, $\sum_{i=1}^{n} \log i \ \leq cn \log n$.

$$\sum_{i=1}^{n} \log i = \log 1 + \log 2 + \cdots + \log n$$
$$\leq n \log n$$

for every $n \geq 1$. Let $c = 1$, $n_0 = 1$. By the definition of O, we have $\sum_{i=1}^{n} \log i$ is $\mathrm{O}(n \log n)$.

Exercise 7

Show that the sum $\sum_{i=1}^{n} \log i$, appearing in the analysis of heap-sort, is $\Omega(n \log n)$.

$$\sum_{i=1}^{n} \log i \geq \sum_{i=\lceil n/2 \rceil}^{n} \log i \geq \sum_{i=\lceil n/2 \rceil}^{n} \log \frac{n}{2} = \left(\frac{n}{2}\right) \log \frac{n}{2}$$

计算这个下界：

$$\left(\frac{n}{2}\right) \log \frac{n}{2} = \frac{n}{2}(\log n - \log 2) = \frac{n}{2} \log n - \frac{n}{2} \log 2 = \Omega(n \log n)$$

Exercise 7: Solution

Proof: To show $\sum_{i=1}^{n} \log i$ is $\Omega(n \log n)$, by the definition of Ω, we need to show that there exist a positive real constant c and a positive integer n_0 such that for all $n \geq n_0$, $\sum_{i=1}^{n} \log i \ \geq cn \log n$.

$\sum_{i=1}^{n} \log i$ = $\log 1 + \log 2 + \cdots + \log n$

$\geq \frac{n}{2} \log \frac{n}{2}$ (consider the last $\frac{n}{2}$ terms)

= $\frac{n}{2} \log n \ - \frac{n}{2}$

Let $c = \frac{1}{4}$. By solving $\frac{n}{2} \log n \ - \frac{n}{2} \geq cn \log n$, we have $n \geq 4$. So we let $n_0 = 4$. By the definition of Ω, we have $\sum_{i=1}^{n} \log i$ is $\Omega(n \log n)$.