

**Exercise 1.** The Selection Sort algorithm sorts the input sequence  $a_1, ..., a_n$  of real numbers in ascending order

## Algorithm 1 Selection Sort

```
1: for i = 1 to n - 1 do
       \min \leftarrow i + 1
 2:
       for j = i + 1 to n do
 3:
          if a_{\min} > a_j then
             \min \leftarrow j
 5:
          end if
 6:
       end for
 7:
 8:
       if a_i > a_{\min} then
         swap a_i and a_{\min}
9:
       end if
10:
11: end for
```

Which sequence will result in the most operations when using Selection Sort (swaps and comparisons)?

 $\bigcirc$  9, 3, 8, 6, 5, 1, 7

4, 7, 2, 9, 5, 6, 3

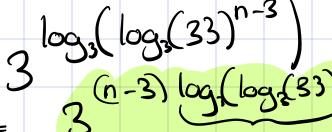
- $\bigcirc 0, 1, 2, 3, 4, 5, 6 \rightarrow$  really fast  $\bigcirc 7, 6, 5, 4, 3, 2, 1 \rightarrow$  really fast

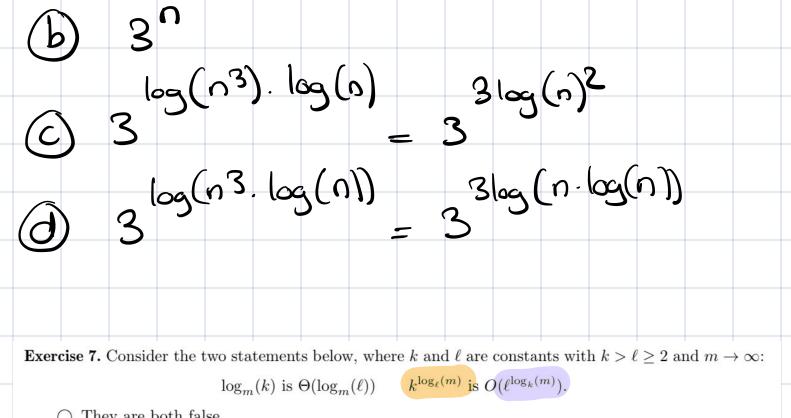
## **Exercise 2.** Which function below grows fastest when n goes to infinity?

- $\bigcirc (\log_3(33))^{n-3}$
- $\bigcirc$  3<sup>n</sup>
- $\bigcap n^{3\log_3(n)}$
- $\bigcap n^3 \log_3(n)$



$$(n-3)$$





- O They are both false.
- Only the first is true.
- Only the second is true.
- O They are both true.

$$\log_{m}(k) = \frac{\log_{\ell}(k)}{\log_{\ell}(m)}$$

$$\log_{m}(\ell) = \frac{\log_{\ell}(\ell)}{\log_{\ell}(m)}$$

Let 
$$c = log_{\ell}(k)$$
.

 $log_{\ell}(m) = log_{\ell}(k) \cdot log_{\ell}(m) \cdot log_{\ell}(m) \cdot c$ 
 $log_{\ell}(m) = log_{\ell}(m) \cdot log_{\ell}(m) \cdot (-c)$ 
 $log_{\ell}(m) = log_{\ell}(m) \cdot log_{\ell}(m) \cdot (-c)$ 

as  $c > 1$ , we  $m \rightarrow \infty$ ,  $m = log_{\ell}(m) \cdot (-c)$ 

**Exercise 9.** Given the two statements below, where d > 0 is an integer constant and  $a_i$  for all  $i \in \mathbf{Z}$  are positive integers with  $\max_{i \in \mathbf{Z}}(a_i) = D$  for a constant D > 0,

$$\sum_{i=0}^{n} a_i i^d \text{ is } \Theta(n^{d+1}) \qquad \qquad \sum_{i=0}^{d} a_i n^i \text{ is } \Theta(n^d)$$

- O They are both true.
- Only the first is true.
- $\bigcirc$  Only the second is true.
- O They are both false.

$$\sum_{i=0}^{n} a_{i}^{i} a_{i}^{i} = 0$$

$$\sum_{i=0}^{n} a_{i}^{i} = 0$$

