Exercise 3 - Solutions

1. Write $n = 2^j$.

   The first pass through the list has $n = 2^j$ array accesses (all other operations have a count of comparable asymptotic order)

   The second pass has $\frac{n}{2} = 2^{j-1}$ array accesses, etc.

   $\Rightarrow$ Total number of array accesses is $A = 2^j + 2^{j-1} + \dots + 2^0 = \dfrac{2^{j+1} - 1}{2 - 1} = O(2^j) = O(n)$

2. Solution with $O(\log n)$-behavior, but needs division by two (i.e. bit-shift operation):

$$m \cdot n = \begin{cases} m & \text{if } n = 1 \\ m\frac{n}{2} + m\frac{n}{2} & \text{if } n > 1,\ n \text{ even} \\ m\frac{n-1}{2} + m\frac{n-1}{2} + 1 & \text{if } n > 2,\ n \text{ odd} \end{cases}$$

   Solution without any division:

$$m \cdot n = \begin{cases} m & \text{if } n = 1 \\ m + m(n-1) & \text{if } n > 1 \end{cases}$$

3. See separate Python code.