0.1 Tutorial 1 - Asymptotic notation

1. (a) true
   (b) false
   (c) true
   (d) false
   (e) false
   (f) false
   (g) true

2. (a) true
   (b) false
   (c) false

3. (a) yes
   (b) no.
   let $T_i(n) = i \cdot n$ and $f(n) = n$

$$\begin{aligned} g(n) &= \frac{n(n+1)}{2} \cdot f(n) \\ &= O(n^2 f(n)) \\ &= O(n^3). \end{aligned}$$

$$\therefore g(n) \neq O(f(n))\text{and } \neq O(n(f(n))).$$

   note: this is because

$$\sum_{i=1}^{n} O(f(n)) \neq O(\sum_{i=1}^{n} f(n)).$$

0.2 Tutorial 2 - Divide and conquer

1. done

2. done

3. find the median. If k > median, then remove all i from A s.t where A[i] < median, vice versa. Time complexity: $\frac{n}{2} + \frac{n}{4} + \ldots + \frac{n}{n} = \frac{(\frac{1}{2})^{\log_2 \frac{n}{2}} - 1}{\frac{1}{2} - 1} \cdot \frac{n}{2} = \frac{\frac{1}{\frac{n}{2}} - 1}{\frac{1}{2} - 1} \cdot \frac{n}{2} = \frac{1 - \frac{n}{2}}{\frac{1}{2} - 1} = O(n)$

4. find the median x_k and then $w = \sum_{i=k}^{n} w_k$. W.L.O.G, if $w > \frac{1}{2}$, then our solution $j > k$. Remove $A[i]$ such that $i < k$. Set $w_k = w_k + 1 - w$. If $w < \frac{1}{2}$, remove all $A[j]$ where $j > k$. Set $w_k = w$. Continue the recursion. The time complexity will be $O(n)$ (analysis the same as Q3)

0.3 Tutorial 3 - Divide and conquer

1. done
2. take a pivot. check which side is larger. Take the larger side. If all elements in one side is the same and len $> \frac{n}{2}$ then we have our answer. Recurse until both sides have len $< n/2$
3. example: 5, -10, -10, 0, 60