Solution 1) This is just a couple examples of what you can do in LaTeX. You may delete it all.

Solution 1a) It isn't considered sufficient to explain your algorithm through pseudocode only, however here is a basic example using the **algorithmicx package**.

Algorithm 1 This squares every element of a list

Require: Integer list of size n

```
function SQUARE(elements)
    for (index, element) $\in \{1, 2, \cdots, n\} \times$ elements do
        list[index] $\leftarrow$ element $\cdot$ element
```

Solution 1b) Algorithm 1 is $O(n)$ because if $T(n)$ is a function which counts the number of $O(1)$ operations, then $T(n)$ is upper bounded by some constant multiple of n, i.e. there exists C such that $T(n) \leq C \cdot n$ for all n large enough.

Solution 1c) For a predicate P, $P(0) \wedge (P(k) \rightarrow P(k+1)) \rightarrow (\forall k \in \mathbb{N}, P(k))$. A function $x : \mathbb{N} \longrightarrow \mathbb{R}$ can be represented as a sequence $\{x_n\}_{n \in \mathbb{N}}$. $\log_2(n) = \log(n)$ in COMP2123!

Solution 2a) Start a new question on a new page each time to make it easier and to hand in on gradescope.