# Solution to Problem 21

**Homework Problem 21.** Give two examples:

- an example of computation time which makes an algorithm feasible according to the formal definition but not practically feasible, and
- an example of computation time for which the corresponding algorithm is practically feasible, but not feasible according to the formal definition.

These examples should be different from what you learned in class – a minor difference is OK.

**Solution.**

First example: $t_A(x) = 10^{2021} \cdot \text{len}(x)$.

- This is a linear function – so it is feasible in the sense of the formal definition.
- However, this number is larger than the number of particles in the Universe, and thus, not practically feasible.

Second example: $t_A(x) = \exp(2^{-2021} \cdot \text{len}(x))$.

- This function is exponentially growing – thus, not feasible in the sense of the formal definition, since every exponential function grows faster than a polynomial.
- However, for every input $x$ of realistic length $\text{len}(x)$, we have

  $$2^{-2021} \cdot \text{len}(x) \ll 1,$$

  thus, $t_A(x) = \exp(2^{-2021} \cdot \text{len}(x)) < \exp(1) = e = 2.71828\ldots$, i.e., $t_A(x) = 1$ or $t_A(x) = 2$, a very feasible number of computational steps.