Programming Languages: FP Practicals 2. Caesar Cipher

Shin-Cheng Mu

Spring 2022

1. Go to our course homepage `https://scmu.github.io/plfp/`. Navigate to the page Syllabus.

2. Download `CaesarCipher.zip`.

3. The main task is to define the functions below:

$$\begin{aligned} &\mathit{encode} :: \mathit{Int} \to \mathit{String} \to \mathit{String} \ , \\ &\mathit{crack} \ \ :: \mathit{String} \to \mathit{Int} \ , \\ &\mathit{decode} :: \mathit{String} \to \mathit{String} \ , \end{aligned}$$

   such that $\mathit{encode}\ k\ \mathit{xs}$ enciphers xs using the key k, $\mathit{crack}\ \mathit{ys}$ takes a ciphered string and tries to recover the key, and $\mathit{decode}\ \mathit{xs}$ deciphers the input string (using crack).

4. Many auxiliary functions are currently given as “undefined”. You may need to define your own auxiliary functions too.

5. This practical is adapted from a chapter in Hutton [Hut07]. For many fascinating stories about cryptography, see Singh [Sin00].

References

[Hut07] Graham Hutton. *Programming in Haskell.* Cambridge University Press, 2007.

[Sin00] Simon Singh. *The Code Book: The Science of Secrecy from Ancient Egypt to Quantum Cryptography.* Anchor, 2000.