1 Lecture

1.1 Foundations

Given a set of characters A, a sequence is a function from $\{1, \ldots, \mathrm{n}\}$ to A (e.g. $\langle a, b, c\rangle = \{(1, a), (2, b), (3, c)\}$). A set of all sequences over A is denoted A^*.

1.2 Featherweight Java

- Notable features: Turing completeness, evaluation order, inheritance.
- Key theorems:
  - FJ programs preserve their types.
  - The only FJ programs that may get stuck are the ones with an invalid downcast in them.
  - FJ programs not containing invalid downcasts are sound.

2 Seminar

1. Implement method `Boolean.and(Boolean that)`.
2. Consider function $value\colon Expression \to Bool$ defined as

$$value(\texttt{new True()}) = true, \tag{1}$$
$$value(\texttt{new False()}) = false. \tag{2}$$

   Prove that

$$\frac{value(e) = v \quad value(e') = v'}{\exists(e'' \in Expression\colon e.\texttt{and(}e'\texttt{)} \Rightarrow^* e'' \wedge value(e'') \Leftrightarrow v \wedge v')}. \tag{3}$$

3. Implement method `Number.minus(Number that)` ($0 - x$ is defined as 0).
4. Prove correctness of your implementation of `Number.minus`. Hint: define function $value : Expression \to N$ and choose carefully what proposition needs to be proved.
5. Use lists to represent numbers in binary format. Implement the following operations: halving, increment and addition.