Theory of computation unit

- Languages, finite automata, and regular expressions (last week Wednesday)
- Equivalence of models of computation (last week Friday)
- The lambda calculus (**today**)
- Computability and tractability (Wednesday)
- Review for Test 3 (Friday)

Today:

- Definition (and motivation) of the lambda calculus
- Building conditionals and control structures
- Building numbers
- Building arithmetic

Test 3: Final exam block, Wednesday, May 6, 1:30 pm

Lambda calculus grammar:

$$
\begin{array}{rcl}
\textit{Expression} & \rightarrow & \textit{Variable} \mid \textit{Application} \mid \textit{Abstraction} \\
\textit{Variable} & \rightarrow & \textit{Identifier} \\
\textit{Application} & \rightarrow & (\textit{Expression}\ \textit{Expression}) \\
\textit{Abstraction} & \rightarrow & \lambda\ \textit{Identifier}\ .\ \textit{Expression}
\end{array}
$$

Lambda calculus evaluate rule:

$$(\lambda x.M\ N) \longrightarrow M[N/x]$$

Emulating multiple parameters (M, N, O, and P could be any expressions.):

$$\begin{aligned}(\lambda x.\lambda y.\lambda z.M)\ N\ O\ P &\longrightarrow \lambda y.\lambda z.(M[N/x])\ O\ P \\ &\longrightarrow \lambda z.(M[N/x,\ O/y])\ P \\ &\longrightarrow M[N/x,\ O/y,\ P/z]\end{aligned}$$

Building booleans and conditionals:

$$\begin{array}{rcl} \mathit{True} & = & \lambda t.\lambda f.t \\ \mathit{False} & = & \lambda t.\lambda f.f \\ \mathit{If} & = & \lambda \ell.\lambda m.\lambda n.((\ell\ m)\ n) \end{array}$$

. . . with the intention that

$$\begin{array}{rcl} \mathit{If}\ \mathit{True}\ M\ N & \longrightarrow_* & M \\ \mathit{If}\ \mathit{False}\ M\ N & \longrightarrow_* & N \end{array}$$

Church numerals:

$$
\begin{aligned}
c_0 &= \lambda z.\lambda s.z \\
c_1 &= \lambda z.\lambda s.(s\ z) \\
c_2 &= \lambda z.\lambda s.(s\ (s\ z)) \\
c_n &= \lambda z.\lambda s.\underbrace{(s\ (s\ \ldots(s}_{n \text{ applications of } s} z)\ldots))
\end{aligned}
$$

$$IsZero = \lambda m.((m\ True)\ (\lambda x.False))$$

$$Plus = \underbrace{\lambda m.\lambda\ n.}_{\substack{\text{take two} \\ \text{numbers}}}\ \overbrace{\lambda z.\lambda s.}^{\substack{\text{construct} \\ \text{a number}}}\ \underbrace{(m\ \overbrace{(n\ z\ s)}^{\substack{\text{apply } s \\ \text{to } z \text{ } n \\ \text{times}}}\ s)}_{\substack{\text{take that result} \\ \text{and apply } s \text{ to it } m \\ \text{more times}}}$$

For Fri, Apr 29:

Review Section 12.8 from DMFP; do Exercises 12.8.(1, 2, 5)