Day 14.

1. Non-determinism 非确定性

The next effect we consider will be *non-determinism*. As before, we extend our language with new term forms characterizing our effect. In this case, the term form we introduce will add a non-deterministic choice between two possible outcomes.

$$t ::= z \mid t \odot t \mid x \mid \lambda x.t \mid t\ t \mid t \ \mathsf{or}\ t$$

Intuitively, the term $1 \ \mathsf{or}\ 2$ should be able to evaluate to either 1 or to 2. We can take advantage of our relational characterization of evaluation to express this quite directly.

$$\frac{t_1 \Downarrow v}{t_1 \ \mathsf{or}\ t_2 \Downarrow v} \qquad \frac{t_2 \Downarrow v}{t_1 \ \mathsf{or}\ t_2 \Downarrow v}$$

That is to say: we now have two different rules for evaluating the same term, each of which could apply in a given derivation. For example, all of the following are valid derivations in this system:

$$\dfrac{\dfrac{}{1 \Downarrow 1}}{1 \ \mathsf{or}\ (2 \ \mathsf{or}\ 3) \Downarrow 1} \qquad \dfrac{\dfrac{\dfrac{}{2 \Downarrow 2}}{2 \ \mathsf{or}\ 3 \Downarrow 2}}{1 \ \mathsf{or}\ (2 \ \mathsf{or}\ 3) \Downarrow 2} \qquad \dfrac{\dfrac{\dfrac{}{3 \Downarrow 3}}{2 \ \mathsf{or}\ 3 \Downarrow 2}}{1 \ \mathsf{or}\ (2 \ \mathsf{or}\ 3) \Downarrow 3}$$

- For the first time, we're relying on an evaluation rule that is *not* functional: we have $\Downarrow \in \mathcal{T} \times V$ but not $\Downarrow \in \mathcal{T} \rightharpoonup \mathcal{V}$
- Our evaluation relation doesn't specify any priorities, preferences, or probabilities. We could imagine implementations of this evaluation relation that produced 1, 2, and 3 with equal likelihood, produced 1 half the time, and 2 and 3 each a quarter of the time, or even always produced 2. Each of these implementations would satisfy the evaluation relation, but would have very different practical uses.

This construct might not seem like much, but is theoretically quite powerful. A couple of examples.

Of course, given binary non-determinism, we can extrapolate to n-ary non-determinism. Here's a general pattern:

$$\mathit{below} = \mathsf{fix}\,(\lambda \mathit{below}.\lambda n.\mathsf{if}\ n = 1\ \mathsf{then}\ 1\ \mathsf{else}\ n\ \mathsf{or}\ \mathit{below}\,(n-1))$$

E.g. $\mathit{below}\,1 \Downarrow 1$, $\mathit{below}\,2 \Downarrow 2$ and $\mathit{below}\,2 \Downarrow 1$. For which z can we derive $\mathit{below}\,0 \Downarrow z$?

Where might we see non-determinism in practice?

Examples: non-determinism and short-cut evaluation. Non-determinism and non-termination.

14.

get 得到当前的state
put 改变当前的状态并且更新当前的值
相当于new一个，一个put等于dereference改值

2. Combining Effects

Suppose we wanted a system with both non-determinism and state?

$$t ::= z \mid t \odot t \mid x \mid \lambda x.t \mid t\ t \mid t\ \mathsf{or}\ t \mid \mathsf{get} \mid \mathsf{put}\ t$$

That's the easy bit... what happens to the evaluation relation? We can try just combining the evaluation relations:

$$\frac{}{\mathsf{get} \mid s \Downarrow s \mid s} \qquad \frac{t \mid s_1 \Downarrow v \mid s_2}{\mathsf{put}\ t \mid s_1 \Downarrow v \mid v} \qquad \frac{t_1 \mid s_1 \Downarrow v \mid s_2}{t_1\ \mathsf{or}\ t_2 \mid s_1 \Downarrow v \mid s_2} \qquad \frac{t_2 \mid s_1 \Downarrow v \mid s_2}{t_1\ \mathsf{or}\ t_2 \mid s_1 \Downarrow v \mid s_2}$$

What about combining them the "other way":

$$\frac{t_1 \mid s_1 \Downarrow v \mid s_2 \quad t_2 \mid s_2 \Downarrow w \mid s_3}{t_1\ \mathsf{or}\ t_2 \mid s_1 \Downarrow v \mid s_3} \qquad \frac{t_1 \mid s_1 \Downarrow v \mid s_2 \quad t_2 \mid s_2 \Downarrow w \mid s_3}{t_1\ \mathsf{or}\ t_2 \mid s_1 \Downarrow w \mid s_2}$$

得到不同的state