We can also provide a translation for recursive terms, $\mu(x)b$, that is more compact than their natural definition as $\langle\!\langle \mathbf{fix}(\lambda(x)b)\rangle\!\rangle$. We set:

$$\begin{aligned}&\langle\!\langle \mu(x)b\{x\}\rangle\!\rangle \triangleq\\ &\quad [rec{=}\varsigma(x)\langle\!\langle b\{x\}\rangle\!\rangle\{\!\{x{\leftarrow}x.rec\}\!\}].rec\end{aligned}$$

and obtain the unfolding property $\mu(x)b\{x\} = b\{\!\{\mu(x)b\{x\}\}\!\}$:

$$\begin{aligned}&\langle\!\langle \mu(x)b\{x\}\rangle\!\rangle\\ &\quad\equiv [rec{=}\varsigma(x)\langle\!\langle b\{x\}\rangle\!\rangle\{\!\{x{\leftarrow}x.rec\}\!\}].rec\\ &\quad= \langle\!\langle b\{x\}\rangle\!\rangle\{\!\{x{\leftarrow}x.rec\}\!\}\{\!\{x{\leftarrow}[rec{=}\varsigma(x)\langle\!\langle b\{x\}\rangle\!\rangle\{\!\{x{\leftarrow}x.rec\}\!\}]\}\!\}\\ &\quad\equiv \langle\!\langle b\{x\}\rangle\!\rangle\{\!\{x{\leftarrow}[rec{=}\varsigma(x)\langle\!\langle b\{x\}\rangle\!\rangle\{\!\{x{\leftarrow}x.rec\}\!\}].rec\}\!\}\\ &\quad\equiv \langle\!\langle b\{x\}\rangle\!\rangle\{\!\{x{\leftarrow}\langle\!\langle \mu(x)b\{x\}\rangle\!\rangle\}\!\}\\ &\quad\equiv \langle\!\langle b\{\!\{\mu(x)b\{x\}\}\!\}\rangle\!\rangle\end{aligned}$$

The recursion operators described use only object primitives. With λ-abstraction and application, we can write the fixpoint operator of Fisher, Honsell, and Mitchell [61]:

$$fix' \triangleq \lambda(f)[rec{=}\varsigma(s)f(s.rec)].rec$$

whose translation is similar, but not identical, to our *fix*:

$$\langle\!\langle fix'\rangle\!\rangle \equiv [arg{=}\varsigma(x)x.arg,\ val{=}\varsigma(x)[rec{=}\varsigma(s)((x.arg).arg{:=}s.rec).val].rec]$$

6.5 Examples

In later chapters we confront the problem of finding useful type systems for the untyped ς-calculus. We now examine some examples that can be easily written in the untyped calculus, but pose interesting typing difficulties. Our examples involve updating through self, and some are based on updating proper methods. The typing requirements range from very basic ones, such as typing geometric points, to very sophisticated ones, such as typing an object-oriented version of the numerals inspired by Scott numerals [124]. For the impatient, typed versions of these examples can be found in Sections 9.4, 15.2, and 16.5, and imperative variants in Sections 10.4 and 17.4.

In these examples we use boolean, integer, and real constants; one of the examples illustrates how to embed basic data types within our calculus. The notation of the untyped λ-calculus is also used freely, since it can be encoded.

6.5.1 Movable Points

A classic object-oriented example involves geometric points with a move method that updates the point coordinates.

$$\begin{aligned}&origin_1 \triangleq\\ &\quad [x = 0,\\ &\quad\ mv\_x = \varsigma(s)\ \lambda(dx)\ s.x := s.x{+}dx]\end{aligned}$$