finite set is recursive. The complement of a recursive set is recursive; for the complement of A is $\neg A$. The union and intersection of two recursive sets is recursive; for the union of A and B is $A \vee B$, and the intersection of A and B is A & B.

Recall that $\forall x$ means for all x and $\exists x$ means for some x. We call $\forall x$ a universal quantifier and $\exists x$ an existential quantifier. As we shall see in §13, these are not recursive symbols. We introduce some modified quantifiers, called bounded quantifiers, which are recursive. We let $(\forall x < y)X(x)$ mean that $X(x)$ holds for all x less than y, and let $(\exists x < y)X(x)$ mean that $X(x)$ holds for some x less than y. To see that these are recursive, note that

$$(\forall x < y)X(x) \leftrightarrow \mu x(\neg X(x) \vee x = y) = y,$$

$$(\exists x < y)X(x) \leftrightarrow \mu x(X(x) \vee x = y) < y.$$

To allow us to use bounded quantifiers with $\leq$ instead of $<$, we agree that $(\forall x \leq y)$ means $(\forall x < y+1)$ and similarly for $\exists$.

We summarize the results of this section. If a function or a relation has an explicit definition or an inductive definition or a definition by cases in terms of recursive symbols, then it is recursive. Recursive symbols include variables, names of recursive functions and relations, μ, propositional connectives, and bounded quantifiers. The recursive functions include the initial functions, $+$, $\cdot$, x^y, $\dot{-}$, and all constant total functions. The recursive relations include all finite relations, $<$, $>$, $\leq$, $\geq$, and $=$.

7. Codes

Suppose that we wish to do computations with a class I other than ω as our set of inputs and outputs. One approach is to assign to each member of I a number, called the code of that member, so that different codes are assigned to different members. Given some inputs in I, we first replace each input by its code. We then do a computation with these numerical inputs to obtain a