rec.1 The Normal Form Theorem

cmp:rec:nft:sec

cmp:rec:nft:thm:kleene-nf

Theorem rec.1 (Kleene's Normal Form Theorem). *There is a primitive recursive relation $T(e, x, s)$ and a primitive recursive function $U(s)$, with the following property: if f is any partial recursive function, then for some e,*

$$f(x) \simeq U(\mu s\, T(e, x, s))$$

for every x.

The proof of the normal form theorem is involved, but the basic idea is simple. Every partial recursive function has an *index* e, intuitively, a number coding its program or definition. If $f(x) \downarrow$, the computation can be recorded systematically and coded by some number s, and the fact that s codes the computation of f on input x can be checked primitive recursively using only x and the definition e. Consequently, the relation T, "the function with index e has a computation for input x, and s codes this computation," is primitive recursive. Given the full record of the computation s, the "upshot" of s is the value of $f(x)$, and it can be obtained from s primitive recursively as well. explanation

The normal form theorem shows that only a single unbounded search is required for the definition of any partial recursive function. Basically, we can search through all numbers until we find one that codes a computation of the function with index e for input x. We can use the numbers e as "names" of partial recursive functions, and write φ_e for the function f defined by the equation in the theorem. Note that any partial recursive function can have more than one index—in fact, every partial recursive function has infinitely many indices.

Photo Credits

Bibliography