The Theory of Computation
Bernard M.E. Moret
Addison-Wesley-Longman, 1998

Solution Manual
Chapter 4, Problem 5.1

The diagonalization in the halting argument of Figure 4.1 comes from running procedure P on arguments x and x. That is, we have, in spirit, tabulated all possible programs (well, almost: see below) and all possible inputs to form a semi-infinite table; then we run procedure P on the diagonal in this table—looking at the xth program run on the xth input.

The assumption that we are trying to disprove here is that P exists—and, in particular, that it runs on all possible programs and all possible inputs. What the argument shows is that P cannot even be fully defined on the diagonal, much less on the whole table—that is, that there cannot exist a procedure P that, for each x, decides whether or not program x run on input x halts.

We do this by sweeping down the diagonal and creating a new program Q on which P must fail to yield an answer. To place it in the perspective of Cantor's approach (creating something that should be in the enumeration but cannot be without causing a contradiction), we can look at a somewhat different table: the table of all programs x such that P correctly decides whether or not program x halts on input x. If P is what we claim it to be, then every single program must appear in the table; but we construct a particular program Q that cannot be in the table—P cannot decide whether Q halts when run on its own code without causing a contradiction. Q is thus the object that should be in the enumeration because of what the enumeration claims to be, but that cannot be in the enumeration without causing a contradiction.