Computation Theory 2003, p 4(11), q 9 AMP 2.

2 Church-Turing Thesis (CTT): every algorithm (in an informal sense) can be realized as a Turing machine.

2 Evidence: other models of computation ~~somfam~~ devised so far (such as various enhancements of TMs, register machines, lambda calculus, Post systems, [description of] partial recursive functions (PR), ...) have all turned out to be algorithmically equivalent

4 to the TM model.

By CTT, there are RMs M_f & M_g that compute f & g. Suffices to construct RM's for h & k from M_f & M_g and appeal to CTT again to see that h & k are PR.

Machine for h:

4 START → [copy R1 to X] → M_f → [Copy X to R1] → M_g → [copy X to R0] → HALT

(where X is a fresh register)

Machine for k (informal algorithm): assuming M_f & M_g use disjoint registers (take a copy of one if not), carry out the steps of computation of M_f & M_g

6 <u>alternately</u>, halting & returning x if either program halts. By CTT, there is a RM realising this informal algorithm.

1 f' is <u>not</u> necessarily PR if f is.

To see this, consider

1
$$f(x) = \begin{cases} 1 & \text{if } x \text{ is code of a RM prog. that eventually halts when started with all registers} = 0 \\ \text{undefined otherwise} \end{cases}$$

Clearly there is an algorithm computing f, so by CTT, it is PR. But if f' were also PR, hence computable by a RM, we
4 could solve the Halting Problem: interleave the steps of computation of f & f', returning 1 if f halts, 0 if f' halts. Since one or other of $f(x)$ & $f'(x)$ is defined (by definition of f'), we get a decision procedure for
6 whether the x^{th} RM eventually halts — which is impossible — so f' could not be PR.

Commentary

The CTT was covered in lecture 7.

Computability of h & k is similar to examples covered in the last 2 lectures of the course (about r.e. sets).

Finding a PR f for which f' is not PR tests intuitive understanding of what is, and is not, effectively computable. Formally speaking, it relies on undecidability of the Halting Problem, covered in first ½ of the course.