Computation Theory 2000

p4q8
JKMM
p11q9

Question 2 Notes Actually rather straightforward, though the use of sequence rather than total function may throw them. The last part requires slight lateral thinking

---

Solution i) given $S \subseteq \mathbb{N}$ define

$\chi_S(n) = (n \in S)$. S is recursive iff χ_S is computable. 3

ii) $S \subseteq \mathbb{N}$ is recursively enumerable iff either $S = \emptyset$ or S is the range of some TOTAL recursive function (either of general arity n or of arity 1, don't care). 2

Two ways to do this. First is to say that given a (countable) set of

Question 2 Solution ctd)

computable functions F say, require that $F = \{f_n\}$, where $\phi(n,x)$ is a computable function and

$$\phi(n,x) = \begin{cases} f_n(x) & \text{when defined} \\ \text{UNDEFINED} & \text{when } f_n(x) \text{ undefined} \end{cases}$$

OR could say that we adopt some coding scheme for all computable functions, and identify F with the set of the codes of the functions belonging to F. Then F is recursively enumerable iff the set of codes is recursively enumerable. Since I can compute given the code this implies the first definition.

3

Question 2 Solution ctd)

The riders are bookwork apart from a slightly lateral flip required for iii)

i) Let $\phi(n, x)$ be a function that lists the characteristic functions

$$\chi_n(x) = \phi(n, x) \text{ of recursive s/s } S_n.$$

Define $\chi(x) = 1 - \phi(x, x)$, certainly the characteristic function of some set T. Now

$$\chi(n) = 1 - \phi(n, n) = 1 - \chi_n(n),$$

hence $T \neq S_n$ for any n.

Hence no recursive enumeration of recursive subsets of $\mathbb{N}$ can be exhaustive.

5

Question 2 Solution ctd)

ii) This is the same all over again!
Suppose given a computable function $\psi(n,x)$ where $S_n(x) = \psi(n,x)$ is an enumeration of the sequences $\{S_n\}$.

Define $t(x) = S(\psi(x,x))$, certainly a TOTAL function, defining a sequence t that is distinct from each S_n. (2)

---

iii) What's needed is a coding scheme for general n-tuplets ... they have one, see page 15 in the notes. Use BINARY

n 1's represent n items,

$$\boxed{..01...001 \quad .. \quad ....0010..00} \qquad n \equiv l_0$$

x_n 0's . . . . x_1 0's

(or backwards) (5)