UG3 Computability and Intractability (2012-2013)
Exercise Sheet 2: Sample Solutions

Submission date: Friday 9 November.

1. (a) The machine shifts the input right by one square, placing a blank symbol on the first square. It leaves the head scanning the last non-blank input symbol (if the input string was not empty).

   Guidelines: Straightforward, award 2 marks if they have the right idea but miss a detail. [*3 marks*]

   (b) The encoding: (using $q_0 = 00$, $q_1 = 01$, $q_2 = 10$, $q_3 = 11$)

   `00001B1*00110B1*0100101*0111001*01B1100*1000111*1011011*10B1110`

   Guidelines: A gift question, all marks or none. [*3 marks*]

   (c) Using C to stand for the above encoding, the configurations of M and the tape of the UTM at the start will be:

   `$000*`C`$^010`

   and at the end it will be:

   `$11Y*XXXXYZY*XXYYXZY*XYXXYXY*XYYYXXY*XYZYYXX*YXXXYYY*YXYYXYY*YXB1110$B0^10`

   The state of M at the end is the one corresponding to the code `11`, i.e., q_3.

   Guidelines: Another gift; award one mark for each bit. If the encoding of the previous part was wrong then run the simulator on that encoding and award full marks if the answer here is correct, else 0. [*3 marks*]

   (d) M takes 4 and the simulation 5295 steps.

   Guidelines: All or none here. [*2 marks*]

2. (a) The reduction is as follows. On an input of the form $\langle M\rangle\$x$ a description of the binary TM M_x is generated, where the behaviour of M_x is as follows:

   On input w:
   if ($w \neq x$) then accept
   else accept iff M halts on x

   Inputs that are not of the form $\langle M\rangle\$x$ are mapped to 'garbage' strings. It is clear that $\langle M\rangle\$x \in L_{\text{halt}} \Leftrightarrow \langle M_x\rangle \in L_{\text{all}}$.

   Guidelines: Award 6 marks for a perfect solution. Interpolate as necessary. [*6 marks*]

(b) Simply note that the above reduction also reduces $\overline{L_{\text{halt}}}$ to $\overline{L_{\text{all}}}$.

Guidelines: Award no more than 1 mark if they base proof on the assumption that L_{all} is r.e. *[5 marks]*

3. (a) We can reduce L_{loop} to L_{odd}. On an input of the form $\langle M\rangle\$x$ a description of the binary TM M_x is generated, where the behaviour of M_x is as follows:

   On input w,
   if the length of w is odd then accept
   else accept iff M halts on x

   Inputs that are not of the form $\langle M\rangle\$x$ are mapped to 'garbage' strings. It is clear that $\langle M\rangle\$x \in L_{\text{loop}} \Leftrightarrow \langle M_x\rangle \in L_{\text{odd}}$.

   Guidelines: Award 6 marks for a perfect solution. Deduct 1 mark for each minor error. Interpolate as you see fit. *[6 marks]*

   (b) $\overline{L_{\text{odd}}}$ either contains strings that are not valid descriptions of TMs or contains strings that describe TMs that accept at least one input of even length. The former type can be identified easily. For the latter, the idea is to simulate the computation of the given input machine M on all strings of even length until one that is accepted by M is found. Care should be taken to ensure that all the computations are performed in parallel in some sense. This can be done by enumerating pairs of the form (x, n) where x is a string of even length and $n \geq 0$ is a natural number and then simulating M on x for n steps. If M ever accepts, then $\langle M\rangle$ is accepted.

   Guidelines: Award 5 marks for a perfect solution. Don't award any marks if the key idea of enumerating pairs (or equivalent) is missing. Be generous. *[5 marks]*

Rahul Santhanam, Thursday 25 October