Problem Set 7

This problem set is due on Friday March 19, by 4:00pm.

Use the CS172 drop box.

Write your name and your student ID number on your solution. Write legibly. The description of your proofs should be as clear as possible (which does not mean long – in fact, typically, good clear explanations are also short.) Be sure to be familiar with the collaboration policy, and read the overview in the class homepage www.cs.berkeley.edu/~luca/cs172.


The definition of an oracle is given in Sipser definition 6.16 on page 211. An oracle is essentially a subroutine. You could interpret this problem as asking for an algorithm that on input $x$, computes the descriptive complexity of $x$, that is, $K(x)$, using a subroutine for $A_{TM}$. On input $\langle M, w \rangle$ the subroutine will return 1 if $M$ accepts $w$, and 0 otherwise. Whenever you invoke the subroutine on some input $\langle M, w \rangle$, use the terminology “query the $A_{TM}$ oracle on input $\langle M, w \rangle$”.

For instance, the machine $S$ in the proof that $HALT_{TM}$ is undecidable in Sipser theorem 5.1 (page 172-173) is an example of an algorithm for $HALT_{TM}$ using an oracle for $A_{TM}$. In that example, $S$ only uses the $A_{TM}$ subroutine once. In general (and for this problem), you are allowed to invoke the subroutine any number of times, and the oracle queries may be adaptive (that is, the next query may depend on the answers to the previous ones).


3. Sipser problem 7.13