Group 1: Dillon Larson, Holly Lind, Jonathan Wong

Problem 9

Definition of Computation History:

Let M be a TM and x an input string.

An accepting computation history for M on x is a sequence of configurations C1,...,Ck, where C1 is the start configuration of M on x, Ck is an accepting configuration of M, and each Ci legally follows from Ci−1 according to the rules of M.

A rejecting computation history for M is defined similarly, except that Ck is a rejecting configuration.

Prove: E_{LBA} is undecidable

The reduction is from A_{TM}. We show if E_{LBA} is decidable then A_{TM} also would be decidable. Suppose R is decision procedure for E_{LBA}. Let

L={w|w is a string of the form C1#C2⋯#Ck giving a legal accepting computation history of M on input x}.

One can show that L can be recognized by an LBA; let's call it B. Further, if L is empty, ⟨M,x⟩ is not in ATM. So if ELBA were decidable the following would be a decision procedure for A_{TM}:

S= "On input ⟨M,x⟩, where M is a TM and x is a string:

Construct LBA B from M on x .

Run R on input ⟨B⟩.

If R rejects, accept; if R accepts, reject." Q.E.D.

In fact, E_{LBA} is Turing-complete for co-r.e. for essentially the same reasons as E_{TM} was.