CS250 Lab 14 – A Universal Turing Machine

Objectives: In this lab you will learn how to

- use a universal TM

A universal TM, U, is a TM that simulates other TMs. By providing a description of all transitions for some TM M and the initial contents of a input tape for M, U can simulate M. The file ex14-universal is a 3-tape machine which implements a universal TM. The first tape holds a description of M's transitions. The second tape hold M's internal tape. The third tape holds M's internal state.

We first establish some conditions on M. M is a single tape machine with one initial state q1 and one final state q2 and we will assume that the blank symbol is symbol a1 in M's alphabet.

We now will describe a method to encode M's tape symbols, states, and transitions as strings of 0's and 1's. Each state q_i in M is encoded as 1^i (for example q4 is encoded as 1111), and each tape symbol a_j is similarly encoded as 1^j. Finally, we encode the move symbols L, S, and R as 1,11, and 111, respectively. The symbol 0 suffixes each string of 1's. So if the internal tape of M is a3 a5 a2 then this will be encoded as 1110111110110.

The encoding of the transitions requires more explanations. Suppose one of M's transitions is $(q_h, a_i) \rightarrow (q_j, a_k, d_l)$ where d1=L, d2=S, and d3=R. This signifies that if M is in state q_h with the tape head on a_i, then M will overwrite a_i with a_k, move the head in the d_l direction, and move to state q_j. Each transition is encoded as $1^h01^i01^j01^k01^l0$ with all encoded transitions concatenated together to form the machine description for tape1. For example, one would encode $(q1, a3) \rightarrow (q2, a5, S)$ as 101110110111110110 within tape1.

How does ex12-universal work? In brief, U executes a continuous loop. First, U performs a linear search in tape1 for a transition that matches the state in tape3 and a symbol at the current head position on tape2. If the current state and symbol do not match the transition, U, proceeds to the next encoded transition in tape1. If the state and symbol match, U changes tape3 to hold the new state and tape2 to overwrite the current symbol with the transition's new symbol, and moves tape2 head in the direction indicated by the transition's direction symbol. Then tape3 is checked to see if it is the halting state (q2) in which case U accepts; otherwise U starts the loop again.

The lengthy encoding makes interesting TMs tedious to encode and input. However, a very simple machine is manageable. The machine ex14.2 is a transducer (it modifies the tape rather than accepting a language) which converts every `a` to `b` and accepts when it reaches the end of the string. Open up this TM and test it.

We can encode the tape alphabet as blank→1, a→11, b→111.
We can encode the transition $(q1, a) \rightarrow (q1, b, R)$ as 101101011101110.
We can encode the second transition as 101011010110.
Therefore the machine description is 101101011101110101011010110 for tape1.

Suppose we want our initial input to be `aaa`. This we encode as 110110110 for tape2. Finally, as always tape3 must hold 1 to signify our initial state q1.

.

Assignment 1:
Perform a Fast Run of the universal machine with the input described above. It should accept. Explain the values on tapes 2 and 3 after the universal machine accepts.

The existence of a universal Turing machine means that TMs can now be considered equivalent to general purpose digital computers which can be programmed to do different jobs at different times!

Submit your files in goucherLearn for grading.