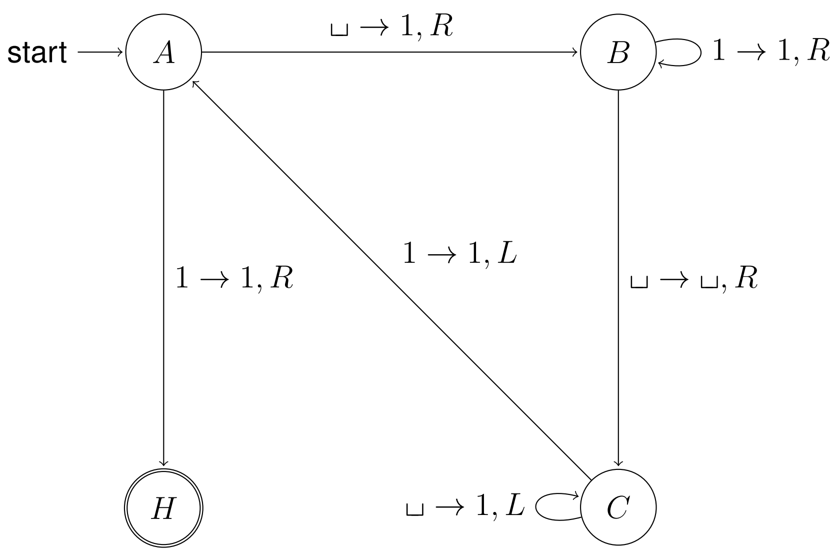
Assignment 3

Due Friday, March 15th, 2019

Submission Instructions: Submit solutions in a single PDF via OWL. Assignments are due at 11:59:59 pm (Eastern Time) on the date listed above. As per the course late policy, assignments submitted more than 48 hours late will not be accepted and a mark of zero (0) will be recorded. Email submissions will not be accepted.

1. [15 marks] Run a Turning Machine

Consider the following description of a Turing machine M.



Run this Turing machine until it halts. Assume the tape is initially empty i.e., every memory location contains $\sqcup$. At each step show:

- The contents of the tape,
- The position of the head,
- The current state.

For example, at the first step $t = 0$, the tape, head and current state are as follows:

								$\downarrow$								
$t = 0$	$\cdots$	$\sqcup$	$\sqcup$	$\sqcup$	$\sqcup$	$\sqcup$	$\sqcup$	$\sqcup$	$\sqcup$	$\sqcup$	$\sqcup$	$\sqcup$	$\sqcup$	$\sqcup$	$\cdots$	State: A

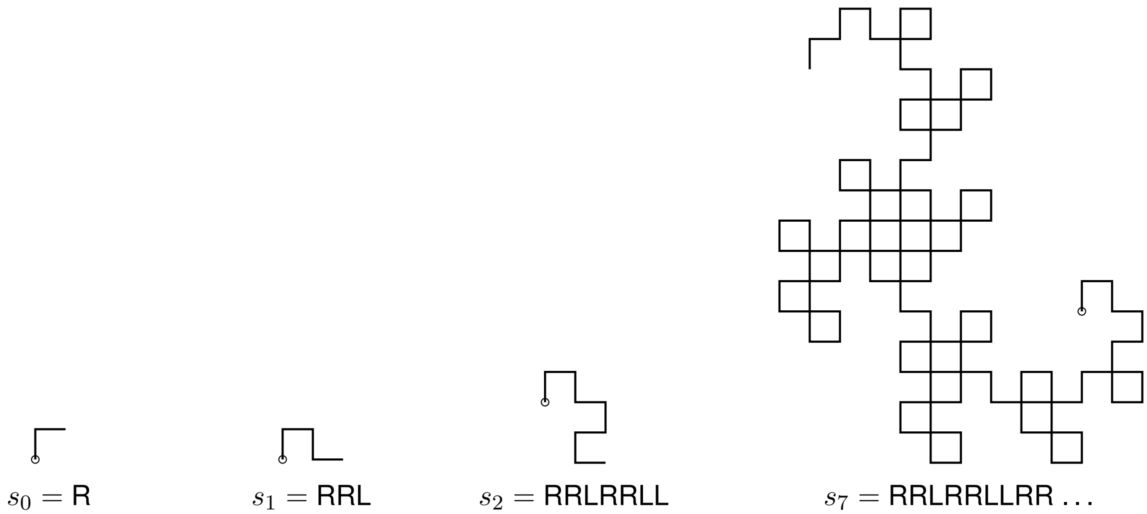


Figure 1: Dragon curve at different iterations.

2. [15 marks] Build a Turing Machine

A Dragon curve is an iterative, self-similar fractal curve. At each iteration i, the curve is defined by a string $s_i \in \{L, R\}^*$ corresponding to a specific sequence of left (L) and right (R) turns (see Figure 1 above). Each successive string s_{i+1} is generted simply by inserting an alternating pattern of R's and L's around the characters of s_i, forming the following sequence:

$$\begin{aligned}
s_0 &= R, \\
s_1 &= RRL, \\
s_2 &= RRLRRLL, \\
s_3 &= RRLRRLLRRRLLRLL, \\
s_4 &= RRLRRLLRRRLLRLLRRRLRRLLLRRLLRLL, \\
&\vdots
\end{aligned}$$

Let $D = \{s_i : i \geq 0\}$ be the set of all Dragon curve iterations as defined above. Draw the state transition diagram of a Turing machine M_D that decides D.

3. [10 marks] Closures of Turing Decidable Langauges

Let M be a Turing machine that decides a language L.

Claim: If L is decidable, then the complement $\overline{L}$ is also decidable.

Proof: Let us define a Turing machine M' that *simulates* the execution of M as a subroutine. We can make M' decide $\overline{L}$ as follows:

On input w:

(a) Run M on w.

(b) If M accepts, halt and output *reject*

(c) Else halt and output *accept*

By the definition of decidability, for all possible input strings w, M will always halt and output either *accept* or *reject* in a finite number of steps. Therefore M' can simulate M's execution to obtain its decision in a finite number of steps—and then return the *opposite* decision. $\square$

Use this approach to show that Turing-decidable languages are:

(a) [5 marks] Closed under union

(b) [5 marks] Closed uner concatenation

4. [10 marks] Closures of Turing Recognizable Langauges

Let Turing machine M recognize a language L. By the definition of recognizability the only requirement is that M run on input w will always halt and output *accept* in a finite number of steps if and only if $w \in L$.

Using an approach similar to the previous question, show that Turing-recognizable languages are:

(a) [5 marks] Closed under union

(b) [5 marks] Closed uner intersection