Too Many Definitions

- Informally, a Turing machine is a machine with a finite set of states, a tape (memory) that is infinite in one direction that can process inputs over some alphabet. At each step, the machine makes the following decisions (based on the state it is in and the symbol it’s tape-head is currently reading): move to some state, write some symbol at the current cell currently under the tape head, and move the tape head to the left or to the right.

- Formally, we define a Turing machine to be a 7-tuple \((Q, q_0, q_{\text{accept}}, q_{\text{reject}}, \Sigma, \Gamma, \delta)\), where \(Q\) is the set of states, \(q_0\) is the start state, \(q_{\text{accept}}\) and \(q_{\text{reject}}\) are the final states, \(\Sigma\) is the input alphabet, \(\Gamma \supseteq \Sigma \cup \{\sqcup\}\) is the tape alphabet, and \(\delta : Q' \times \Gamma \rightarrow Q \times \Gamma \times \{L, R\}\), where \(Q' = Q \setminus \{q_{\text{accept}}, q_{\text{reject}}\}\) is the transition function.

- A Turing machine is called a **decider** if for all inputs \(x \in \Sigma^*\), it halts and either accepts or rejects \(x\).

- A language \(L \subseteq \Sigma^*\) is called **decidable** if there exists a decider Turing machine \(M\) such that \(L = L(M)\).

- Let \(L\) and \(K\) be languages, where \(K\) is decidable. We say that solving \(L\) reduces to solving \(K\) (or simply, \(L\) reduces to \(K\), denoted \(L \leq K\)), if we can decide \(L\) by using a decider for \(K\) as a subroutine (helper function).
Closure Ceremony
Suppose that $L_1$ and $L_2$ are decidable languages. Show that the languages $L_1 \cdot L_2$ and $L_1^*$ are decidable as well.\footnote{Exercise: show that $L_1 \cup L_2$ and $L_1 \cap L_2$ are also decidable.}

Even Only Please
Prove that the following language is decidable by reducing it to $\text{EMPTY}_{\text{DFA}}$:
\[
\text{NO-ODD-ONES} = \{ \langle D \rangle : D \text{ does not accept any string containing an odd number of 1's} \}.
\]

Only $19.99!$ Call now!
Dr. Hyper Turing Machines Inc LLC is selling a whole host of new Turing machines, each for $19.99$:
- Super-scalable TMs - choose any fixed number of tapes heads you like!
- Quad-core TMs - now with 4 tapes (each with its own tape head)

Normal TMs usually go for $9.99$ these days. Your friend (who’s not very Turing-savvy) is in the market for a new Turing machine and just texted you asking you for purchasing advice. Your instincts tell you that maybe most of this is marketing hype. But some of those improvements do sound pretty compelling... Your friend doesn’t use his TM for all that much - mostly just browsing the web and checking email. What should you recommend he do?

Recognize, Enumerate, Decide
Define a language $A \subseteq \Sigma^*$ to be Turing-recognizable if there is a TM $M$, not necessarily a decider, such that $A = L(M)$. That is, for inputs $x \in A$, $M$ halts and accepts $x$, and for inputs $x \notin A$, $M$ either halts and rejects $x$, or does not halt.

(a) Prove that a language $A$ is decidable if and only if both $A$ and $\overline{A}$ are Turing-recognizable.

(b) Define the language
\[
\text{ACCEPTS}_{\text{TM}} = \{ \langle M, x \rangle : \text{TM $M$ on input $x$, halts and accepts} \}.
\]
Prove that $\text{ACCEPTS}_{\text{TM}}$ is Turing-recognizable, but its complement is not Turing-recognizable.

(c) (Extra) An enumerator is a TM that when started with a blank input tape, outputs a list of strings, possibly with repetition, on a second output tape. An enumerator may run forever if it outputs an unbounded number of strings.
Prove that a language $A$ is Turing-recognizable if and only if there is an enumerator $E$ that only outputs strings that belong to $A$, and for every $x \in A$, $E$ eventually outputs $x$.

(Extra) Not Just Your Regular Old TM
Suppose we change the definition of a TM so that the transition function has the form $\delta : Q \times \Gamma \rightarrow Q \times \Gamma \times \{R, S\}$ where the meaning of $S$ is “stay”. That is, at each step, the tape head can move one cell to the right or stay in the same position. Suppose $M$ is a TM of this new kind, and suppose also that $M$ is a decider. Show that $L(M)$ is a regular language.