Assn 1.

1) Many solutions work (naturally). A nice one is

\[ f(x) = \tan(\pi x + \frac{\pi}{2}) \]

As \( x \to 0^+ \), \( f(x) \to \tan(-\frac{\pi}{2}) \) \( (-\infty) \)

As \( x \to 1^- \), \( f(x) \to \tan(\frac{\pi}{2}) \) \( (+\infty) \)

(and continuous)

\( f(x) \) is strictly increasing in \((0,1)\). Thus \( f \) is a bijection from \((0,1)\) to \(\mathbb{R}\).

2) It is convenient to give a bijection between \((0,1)^\ast\) to \((0,1)\). Use previous problem to get bijection from \(\mathbb{R}^+\) to \(\mathbb{R}\).

We use the interleaving trick.

\[ f(0.x_1x_2x_3\ldots, 0.y_1y_2y_3\ldots) = 0.x_1y_1x_2y_2 \ldots \]

It is routine to show \( f \) is a bijection from \((0,1)^\ast\) to \((0,1)\).

3) Let us construct a bijection from \(S\) to \(\mathbb{N}\).

For any subset \(A \subseteq \mathbb{N}\) and \(k \in A\),

\[ \text{define } \text{rank}_A(k) = \left\lfloor \frac{k}{|\left\{ j \leq k \mid j \in A \right\}|} \right\rfloor \]

Thus, \( \text{rank}_A(k) \) is the index of \( k \) in sorted order of \( A \).

Note that this is well-defined for infinite \( A \).

We assumed an injection \( f : S \to \mathbb{N} \).
Define \( g = \text{rank}_{\text{image}(f)} (f(x)) \)

Thus \( g(x) \) is the rank of \( f(x) \) in the image of \( f \).

Note that \( g \) is also an injection. Furthermore, if \( \exists r \in \mathbb{N} \) and \( x \in S \) st. \( g(x) = r \), then for all \( r' < r \), \( \exists y \in S \) st. \( g(y) = r' \).

(This is a direct property of the rank.)

Thus, the image of \( g \) is a contiguous set of \( \mathbb{N} \), starting from \( \max_{x \in S} g(x) \).

Because \( \max_{x \in S} g(x) \) cannot exist (\( g \) is infinite and \( f \) is injection), the image of \( g \) must be \( \mathbb{N} \).

Thus, \( g \) is a bijection from \( S \) to \( \mathbb{N} \).

4) Standard diagonalization.

We proved that \( \mathbb{Q} \) is countable.

Thus, we can enumerate rationals \( q_1, q_2, \ldots \).

Let us define \( q_i(j) \) to be the \( j \)th bit in \( q_i \) after the binary point.

Define \( n \) such that the \( j \)th bit of \( n \) is \( \neg q_i(j) \).

\( n \) must be irrational.

5) Such a machine can compute "any" \( f: \mathbb{N} \rightarrow \{0,1\} \).

Thus, it can compute the halting problem and is strictly more powerful than TMs.

Simply do the following: For any \( f \):

Make machine \( M_f \) that loads input into natural number register. It applies \( f \) as transition function to more into accept/reject state.
6) We encode all symbols using $\{0,1,\}^3$.

First, using conversions discussed in class, we can assume alphabet is $\{0,1,\}^3$.

Now, we convert as follows:

$$
0 \rightarrow 10 \\
1 \rightarrow 11 \\
\_ \rightarrow 00
$$

Thus, we use two symbols (a pair) to encode an "old" symbol. Any old transition is a single step of the old machine becomes two steps (to write two symbols) of new machine.

4) We can use diagonalization to prove $L_u$ is undecidable, as in the halting problem proof.

It is easier to just reduce the halting problem to $L_u$.

Given machine $M$, construct $M'$ as follows. Simply make $M$ enter into infinite loop if it reaches the reject state.

Thus $M$ accepts $\text{OOPS! Reduction in wrong direction!!}$

Construct following machine. Let's assume $L_u$ is decided by $M_u$. $M_u(M,M)$

1. Convert $M$ to $M'$ by replacing rejecting state with a transition to accept state.

2. Call $M_u(M',x)$ and accept if it does.
   Reject otherwise.
\[ M \xrightarrow{\text{acc}} \text{became} \xrightarrow{\text{qRj}} M' \xrightarrow{\text{Acc}} q \]

M halts on x iff \( M' \) accepts x. Thus \( M_h(\langle M, x \rangle) \) accepts iff \( M_h(\langle M', x \rangle) \) accepts.

Chaining together, \( M_h(\langle M, x \rangle) \) accepts iff M halts on x. Contradiction.

8) We create a machine that explicitly fetches the symbol at the address.

On going to access, the machine:
1. Stores current Marks current position by adding *.
2. Goes to start of tape.
3. Walks the right number of cells to get to location in address tape.
4. Stores symbol in its state (or additional constant sized register).
5. Goes back to position with *.
   
   The run time of a single access is at most, the largest possible address, which is at most \( T \) (in \( T \) steps, the TM can only write to \( T \) cells). Thus, the total time is at most \( O(T^2) \).