Reminder

- One convenient definition.

**Definition 1** A language $A$ is co-Turing-Recognizable if $\overline{A}$ is Turing-Recognizable.

- Mapping Reductions: $A \leq_m B$ if there is a computable function $f: \Sigma^* \rightarrow \Sigma^*$, where for every $w$,

$$w \in A \iff f(w) \in B.$$ 

In picture,

```
A -----> B
     ^   |
     f   |
     |   v
     B -----> A
```

- Using mapping reductions:
  - To show that $B$ is not Turing-Recognizable, show that $\overline{A_{TM}} \leq_m B$.
  - To show that $B$ is not co-Turing-Recognizable, show that $A_{TM} \leq_m B$.

- Summary, in a diagram, of our understating of the computation power of the different models we saw:
Note that the intersection of the Turing-Recognizable languages and the co-Turing-Recognizable languages is exactly Turing-Decidable languages. This is due to the theorem we saw in class that $A$ is decidable iff both $A$ and $\overline{A}$ are Recognizable.

**Example 1 — Two Tape Turing Machine**

Let

$$2\text{TAPE} = \left\{ \langle M, w \rangle \mid M \text{ is a two-tape TM that writes a nonblank symbol in its second tape when it is run on } w \right\}$$

Show that $2\text{TAPE}$ is recognizable and undecidable.

**Solution**

**$2\text{TAPE}$ is recognizable.** To show that $2\text{TAPE}$ is recognizable we construct a TM $M$ that recognize it.

$M = \text{“On input } \langle M, w \rangle, \text{ where } M \text{ is a two tape TM and } w \text{ is any string:}$$

1. Run $M$ on $w$.
2. If during its run, $M$ writes a non-blank symbol on its second tape, $accept$.
3. Otherwise, $reject$.

**$2\text{TAPE}$ is undecidable.** We show that $A_{TM}$ reduces to $2\text{TAPE}$. Assume towards a contradiction that a TM $R$ decides $2\text{TAPE}$. Then construct TM $S$ that uses $R$ to decide $A_{TM}$.

$S = \text{“On input } \langle M, w \rangle, \text{ where } M \text{ is a TM and } w \text{ is any string:}$$

1. Construct the following two-tape TM $T$.
2. Simulate $M$ on $x$ using the first tape.
3. If $M$ accepts, write a non-blank symbol on the second tape.”
2. Run \( R \) on \( \langle T, w \rangle \).
3. If \( R \) accepts, accept. If \( R \) rejects, reject."

We can write this reduction as a mapping reduction \( A_{TM} \leq_m 2\text{TAPe}: f(\langle M, w \rangle) = \langle T, w \rangle \).

**Example 2 — The Infinity Language**

Let

\[
INF_{TM} = \{ \langle M \rangle \mid M \text{ is a Turing machine and } |L(M)| = \infty \}.
\]

Show that \( INF_{TM} \) is not Turing-Recognizable and not co-Turing-Recognizable.

**Solution**

\( INF_{TM} \text{ is not co-Turing-Recognizable.} \) We show \( A_{TM} \leq_m INF_{TM} \). We do so using the same reduction we used in class for the language \( E_{TM} \), namely the machine \( M_w \). Recall the given \( M \) and \( w \), we defined

\( M_w = \text{“On input } x:\) \\
1. Erase \( x \) from the tape. \\
2. Simulate \( M \) on \( w \). \\
3. If \( M \) accepts, accept. \\
4. If \( M \) rejects, reject.”

The mapping reduction is now \( f(\langle M, w \rangle) = \langle M_w \rangle \). Clearly, \( f \) is computable.

- If \( M \) accepts \( w \), then \( L(M_w) = \Sigma^* \), and thus \( |L(M_w)| = \infty \).
- If \( M \) does not accept \( w \), then \( L(M_w) = \emptyset \), and thus \( |L(M_w)| = 0 \neq \infty \).

\( INF_{TM} \text{ is not Turing-Recognizable.} \) We show \( \overline{A_{TM}} \leq_m INF_{TM} \). This is a more challenging task. We need to come up with a machine that has a infinite language when \( M \) does not accept \( w \). We use the technique of “controlled execution” (or “timed execution”), where we simulate a machine for finite amount of steps. Only this time, the number of steps we simulate will depend on the input. Define

\( T = \text{“On input } x:\) \\
1. Simulate \( M \) on \( w \) for \( |x| \) of steps. \\
2. If \( M \) accepts, reject. \\
3. If \( M \) does not accept, accept.”

The mapping reduction is now \( f(\langle M, w \rangle) = \langle T \rangle \). Clearly, \( f \) is computable.

- If \( M \) accepts \( w \), then it does so after finite number of steps, say \( n \). Now, every input \( x \) with \( |x| \geq n \), \( T \) will reject. Since there are only finite number of strings of length at most \( n \), \( L(T) \) is finite and \( \langle T \rangle \notin INF_{TM} \).
- If \( M \) does not accept \( w \), then \( L(T) = \Sigma^* \), and thus \( |L(T)| = \infty \).