The language \textsc{Half} is the set of all Turing machines that compute one half to infinite precision (i.e.
that print 100000...).

\[
\text{\textsc{Half}} = \{ \langle M \rangle \mid M \text{ is a Turing machine that computes } 100000...\}
\]

Prove that \textsc{Half} is undecidable.

**Claim:** \textsc{Half} is undecidable

**Proof by Contradiction.**

Suppose, for the sake of contradiction, that \textsc{Half} is decidable. Then there is a decider \(D_{\text{\textsc{Half}}}\) that decides \textsc{Half}.

By the definition of a decider, if \(M\) is a Turing machine that computes 100000..., then \(D_{\text{\textsc{Half}}}\) accepts on input \(\langle M \rangle\). If \(M\) is a Turing machine that does not compute 100000..., then \(D_{\text{\textsc{Half}}}\) rejects on input \(\langle M \rangle\). There are no other cases since every machine either computes 100000..., or it does not.

Recall that the language \textsc{InfinitePrint}_0 is the set of all machines that output an infinite number of zeroes:

\[
\text{\textsc{InfinitePrint}}_0 = \{ \langle M \rangle \mid M \text{ outputs an infinite number of zeroes}\}
\]

As we showed in class, \textsc{InfinitePrint}_0 is undecidable: it is impossible to create a Turing machine that decides \textsc{InfinitePrint}_0. This will be the basis of the contradiction.

Consider the machine \(D_{\text{\textsc{InfinitePrint}}_0}\) on input \(\langle M \rangle\):

1. Construct the Turing machine \(M'\) as follows:
   - (a) Output a 1
   - (b) Move one cell to the right
   - (c) Simulate \(M\):
     i. Every time \(M\) outputs a digit \(c\):
        - A. If \(c\) is 0: Output a 0
        - B. If \(c\) is 1: Do nothing

2. Run \(D_{\text{\textsc{Half}}}\) on input \(\langle M' \rangle\):
   - (a) If \(D_{\text{\textsc{Half}}}\) accepts, ACCEPT
   - (b) If \(D_{\text{\textsc{Half}}}\) rejects, REJECT
Observe that $D_{\text{InfinitePrint}_0}$ halts on all inputs.

Step 1 simply constructs a Turing machine, which can be done in a finite amount of time. $M'$ is never run, it is simply built.

Step 2 runs $D_{\text{HALF}}$ with the input $\langle M' \rangle$. $D_{\text{HALF}}$ is assumed to be a decider, so it halts in a finite amount of time (on any input).

We claim that $D_{\text{InfinitePrint}_0}$ is a decider for InfinitePrint$_0$.

**Case analysis:** For any machine $M$, there are two cases: $\langle M \rangle$ is in InfinitePrint$_0$ or it is not in InfinitePrint$_0$. Suppose that we run $D_{\text{InfinitePrint}_0}$ on input $\langle M \rangle$:

**Case 1:** $\langle M \rangle \in \text{InfinitePrint}_0$: Because $\langle M \rangle$ is in InfinitePrint$_0$, it outputs an infinite number of zeroes.

$D_{\text{InfinitePrint}_0}$ first constructs $M'$. If we were to run $M'$, it would output a 1, and then output an infinite number of zeroes (because $M$ outputs an infinite number of zeroes). Therefore, $\langle M' \rangle$ is in HALF.

In step 2, $D_{\text{HALF}}$ is run on input $\langle M' \rangle$, and it accepts because $\langle M' \rangle$ is in HALF. Then, $D_{\text{InfinitePrint}_0}$ accepts.

**Case 2:** $\langle M \rangle \notin \text{InfinitePrint}_0$: Because $\langle M \rangle$ is not in InfinitePrint$_0$, it outputs a finite number of zeroes.

$D_{\text{InfinitePrint}_0}$ first constructs $M'$. If we were to run $M'$, it would output a 1, and then output a finite number of zeroes (because $M$ outputs an finite number of zeroes). Therefore, $\langle M' \rangle$ is not in HALF.

In step 2, $D_{\text{HALF}}$ is run on input $\langle M' \rangle$, and it rejects because $\langle M' \rangle$ is not in HALF. Then, $D_{\text{InfinitePrint}_0}$ rejects.

For any machine $M$, $D_{\text{InfinitePrint}_0}$ accepts if $\langle M \rangle$ is in InfinitePrint$_0$ and $D_{\text{InfinitePrint}_0}$ reject if $\langle M \rangle$ is not in InfinitePrint$_0$. Therefore, $D_{\text{InfinitePrint}_0}$ decides InfinitePrint$_0$, so InfinitePrint$_0$ is decidable.

However, this is a contradiction, as InfinitePrint$_0$ is known to be undecidable.

Therefore, our initial assumption must be incorrect. HALF is indeed undecidable. □