Problem 1
Let
\[ \text{EPS}_{TM} = \{ \langle M \rangle \mid M \text{ is a TM that accepts } \varepsilon \}. \]
Show that \( \text{EPS}_{TM} \) is undecidable by giving a reduction from \( \text{ATM} \). You may do this with functions (like we did for, e.g., \( \text{REGULAR}_{TM} \) when we reduced \( \text{ATM} \) to \( \text{REGULAR}_{TM} \), see slide 3 of Section 5.1), or with a picture (like we did for \( \text{EQ}_{TM} \) when we reduced \( \text{EQ}_{TM} \) to \( E_{TM} \), see slide 4).

Problem 2
Read ahead about the Post Correspondence Problem (Section 5.2 – or wait until Tuesday). Use the undecidability of the PCP problem to show that the following language is also undecidable:
\[ \text{INT}_{CFG} = \{ \langle G_1, G_2 \rangle \mid G_1, G_2 \text{ are CFG’s and } L(G_1) \cap L(G_2) \neq \emptyset \} \]
In particular, for a given PCP instance, create \( G_1 \) and \( G_2 \) such that \( L(G_1) \cap L(G_2) \) is nonempty if and only if the PCP instance has a solution.

Problem 3
An unrestricted grammar is like a CFG, except that you can now have any string of variables and terminals containing at least one variable on the left-hand side of a rule. Surprisingly, unrestricted grammars generate exactly the Turing-recognizable languages. We will not show this, but we will look at an example.

The following grammar generates the language \( \{a^{2k} \mid k \geq 1\} \).
\[
\begin{align*}
S & \rightarrow ACaB \\
Ca & \rightarrow aaC \\
CB & \rightarrow DB \\
CB & \rightarrow E \\
aD & \rightarrow Da \\
AD & \rightarrow AC \\
aE & \rightarrow Ea \\
AE & \rightarrow \varepsilon
\end{align*}
\]
Give a derivation of \( aaaaaaaaaa \).
Problem 4

Consider the Hamiltonian cycle problem where, for a given undirected graph, one needs to decide if there exists a cycle that goes through every vertex exactly once. Formally:

\[ \text{HAMCYCLE} = \{ \langle G \rangle \mid G \text{ is a graph that contains a cycle going through every vertex} \}. \]

A similar problem is the Longest cycle problem:

\[ \text{LONGCYCLE} = \{ \langle G, k \rangle \mid G \text{ is a graph that contains a cycle of length } \geq k \text{ (no repeated vertices)} \}. \]

(a) Imagine that you have a black box (function) that solves LONGCYCLE. How would you use it to solve HAMCYCLE?

(b) Approximate the number of steps you need for the conversion(s) you are using – in particular, can you do them in a polynomial number of steps (in the number of vertices and edges)?