(1) Consider the language

\[ L = \{ \langle G_1, G_2 \rangle \mid G_1, G_2 \text{ are context-free languages and } L(G_1) = L(G_2) \} \]

(a) Show that \( L \) is undecidable.
(b) What is \( L \)? Show that \( L \) is recognizable.
(c) Show that \( L \) is unrecognizable.

(2) Show that the following language is in NP.

\[ \text{GRAPH-ISOMORPHISM} = \{ \langle G_1, G_2 \rangle \mid G_1 \text{ and } G_2 \text{ are isomorphic graphs} \} \]

Two graphs \( G_1 = (V_1, E_1) \) and \( G_2 = (V_2, E_2) \) are isomorphic if there is a bijection \( \varphi : V_1 \rightarrow V_2 \) mapping vertices of \( G_1 \) to vertices of \( G_2 \), so that edges and non-edges are preserved, that is \( (u, v) \in E_1 \) if and only if \( (\varphi(u), \varphi(v)) \in E_2 \).

(3) In class, we mentioned two definitions of NP. According to the first definition, a language \( L \) is in NP if it has a polynomial time verifier \( V \). In other words,

\[ x \in L \text{ if and only if there exists } s \text{ such that } V \text{ accepts } \langle x, s \rangle. \]

According to the second definition, a language \( L \) is in NP if it is accepted by a non deterministic polynomial time Turing machine. Here a nondeterministic Turing machine accepts an input \( x \) if it accepts \( x \) in at least one computation path, and such a machine is polynomial time if all of its computation paths have length bounded by the same polynomial.

Show that these two definitions are equivalent. Hint: How is the “candidate solution” \( s \) related to the computation paths in a nondeterministic Turing machine \( M \)?