Languages:
\[ A_{DFA} = \{ < D, w > \mid D \text{ is a DFA, } w \text{ is a string, and } D \text{ accepts } w \} \]
\[ E_{DFA} = \{ < D > \mid D \text{ is a DFA, and } L(D) = \emptyset \} \]
\[ EQ_{DFA} = \{ < M_1, M_2 > \mid M_1 \text{ and } M_2 \text{ are both DFAs, } \]
\[ \text{and } L(M_1) = L(M_2) \} \]
\[ A_{REX} = \{ < R, w > \mid R \text{ is a regular expression, and matches string } w \} \]
\[ E_{REX} = \{ < R > \mid R \text{ is a regular expression and } L(R) = \emptyset \} \]
\[ EQ_{REX} = \{ < R_1, R_2 > \mid R_1 \text{ and } R_2 \text{ are regular expressions, } \]
\[ \text{and } L(R_1) = L(R_2) \} \]
\[ A_{NFA}, E_{NFA}, EQ_{NFA} \text{ defined similarly, but for NFAs.} \]

What about Turing machines?
\[ L_{TM} = \{ < M > \mid M \text{ is a Turing machine} \} \]
\[ A_{CFG} = \{ < G, w > \mid G \text{ is a grammar, } w \text{ is a string, } G \text{ generates } w \} \]
\[ E_{CFG} \text{ and } EQ_{CFG} \text{ are defined similarly.} \]
\[ A_{PDA}, E_{PDA}, EQ_{PDA} \text{ also defined similarly.} \]

A countable language