Runtime of \[ \text{MaxSAT} \]?
Big Picture?
How would we need to change proof to show \[ \text{TOBF} \] is \[ \text{NPSPACE} \] hard?

"\[ y = f(x, c) \] now is a formula that checks if \( x \) results from \( c \) from other monolithic check"

L \[ \in \text{NPSPACE} \] \[ \Rightarrow \]

\[ \text{L Solver} \]

\[ x \in L \]

\[ \text{Pay Stack} \rightarrow \]

\[ \text{Pay Stack} \rightarrow \text{TOBF} \]

\[ \text{Pay Stack} \rightarrow \text{L Solver} \]

\[ \text{Colapse!} \]

**Reductions Review (Proving L \[ \in \text{Class-Hard} \]):**

1. **Direct:** \( L \in \text{Class}, \exists \text{ TM } M \text{ w/ Property } A \)
   and can use transition function \( f \) of \( M \) to create conversion function \( f \) from \( L \) input to \( L' \) input
   \[ x \in L \]

2. **Boost-Step:** Know \( L' \) is Class-Hard.
   \[ \text{Create conversion function } f \text{ from } L' \text{ input to } L \text{ input} \]
   \[ x \in L' \]

**3-SAT: \[ \Pi(x) \]:** \( x \) describes a 3-SAT instance where each variable \( u_i \) appears in at least 3 clauses

**Prove 3-SAT is NP-Hard**

\[ x = (u_1, u_2, \ldots, u_n) \]

\[ x \in \text{3SAT} \]

\[ \text{Don't figure out!} \]

Create \( f \) to turn \( x \) into 3-SAT instance

\[ (u_1, u_2, u_3) \land (u_1, u_2, u_4) \land (u_1, u_2, u_5) \land \ldots \]

\[ (u_1, u_2, u_3) \land (u_1, u_2, u_4) \land (u_1, u_2, u_5) \land \ldots \]

- Replace \( f \)th occurrence of \( u_i \) with \( u_{ij} \) poly-time
- For each \( u_i \) except last include clauses poly-time

\[ (u_{ij}, u_i, u_j) \land (u_{ij}, u_i, u_j) \land \ldots \]

- If \( x \in \text{3SAT} \), \( \exists (u_1, u_2, \ldots, u_n) \) a satisfying assignment
  - Then \( u_{ij} = u_j \) will satisfy \( f(x) \).
  - If \( f(x) \in \text{3SAT} \), all \( u_{ij} \) must have same value.
  - Set \( u_i \) to have that value poly-time satisfying assignment for \( x \).