Forge: Relational Expressions 1

How do we verify that someone who went to Brown always supervises someone who went to Harvale?
- Look at all the possible worlds generated and see if it's true (a **lot** of possible worlds!)
- Ask Forge to show counterexamples to our counterexample!
  - If it can't find counterexamples, our conjecture holds (up to bounds)

We `run {puzzle somebody}` to get all possible worlds in which both predicates are satisfied.

We want to find a word which satisfies `puzzle` but doesn't satisfy `somebody`; if such a world doesn't exist, we can verify that someone from Brown always supervises someone from Harvale
- This returns `UNSAT`; forge has proven that given this supervision relation, it must always hold that someone from Brown supervises someone not from Brown.

Note: running `for 10` gets you up to 10 (use `for exactly 10` to get exactly 10)

Difference between some and all:
- Writing `some n1: Node | all n2: Node | n1 in n2.edges` means there is a hub that everyone feeds into
- Writing `all n1: Node | some n2: Node | n1 in n2.edges` means that every node has an edge coming in
- `^edges` is the transitive closure of all edges (see next lecture)