

Generalized Super Mario Bros

Vorapat Nicklamai 5919369

MARIO
000000

● × 00

WORLD
1-1

TIME

SUPER MARIO BROS.

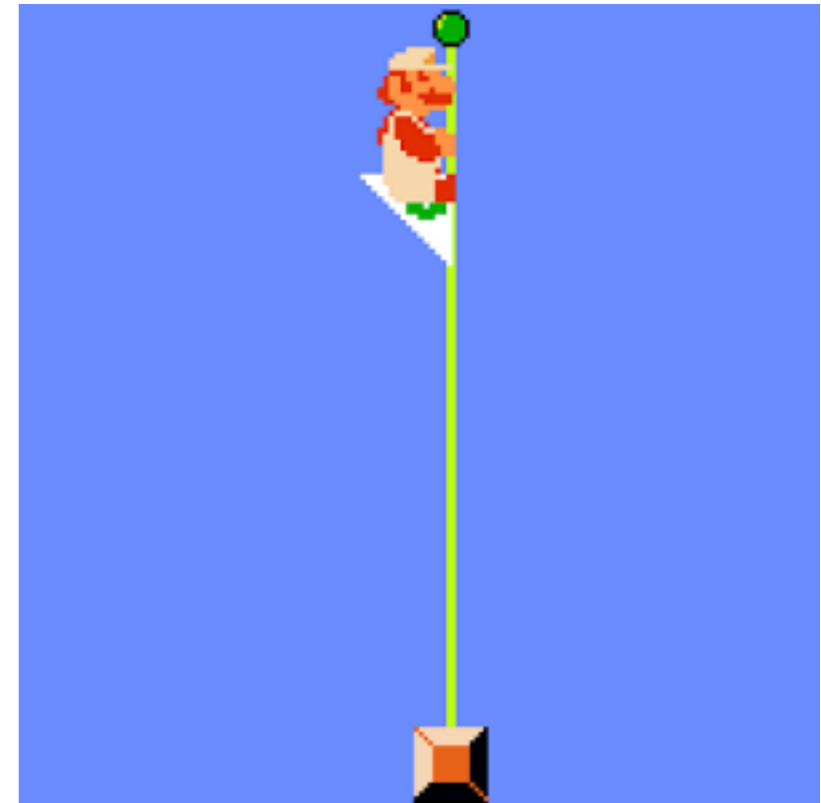
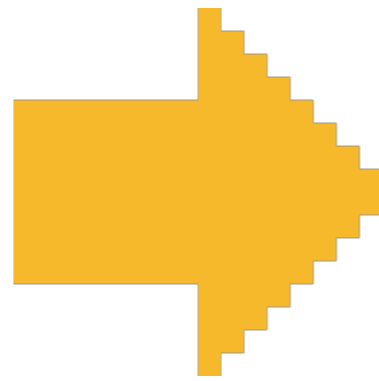
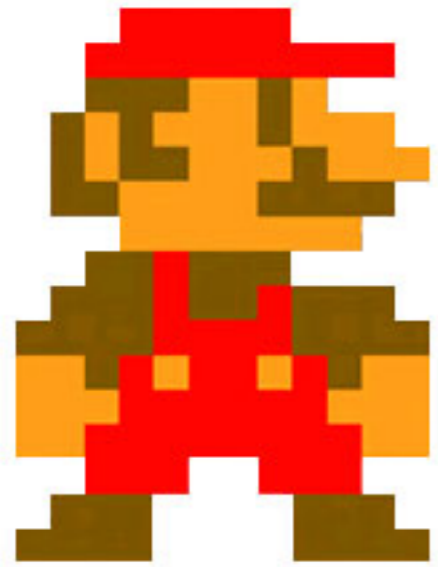
©1985 NINTENDO

- 1 PLAYER GAME
- 2 PLAYER GAME

TOP - 000000



Goal



Proving NP-Completeness

- The problem must be NP
- Other NP-Problem can be reduced to the problem

Is it in NP?

- Non-Deterministic Polynomial Time
- Decision -> True or False
- Guess until accepting a branch (Non-Deterministic Turing Machine)

Choices



3-SAT

- Assign variables such that all clauses are true
- $(x_1 \vee x_2 \vee x_3) \wedge (x_4 \vee x_5 \vee x_6) \wedge (x_7 \vee x_8 \vee x_9)$
- x_i stands for variable
- $()$ stands for clause

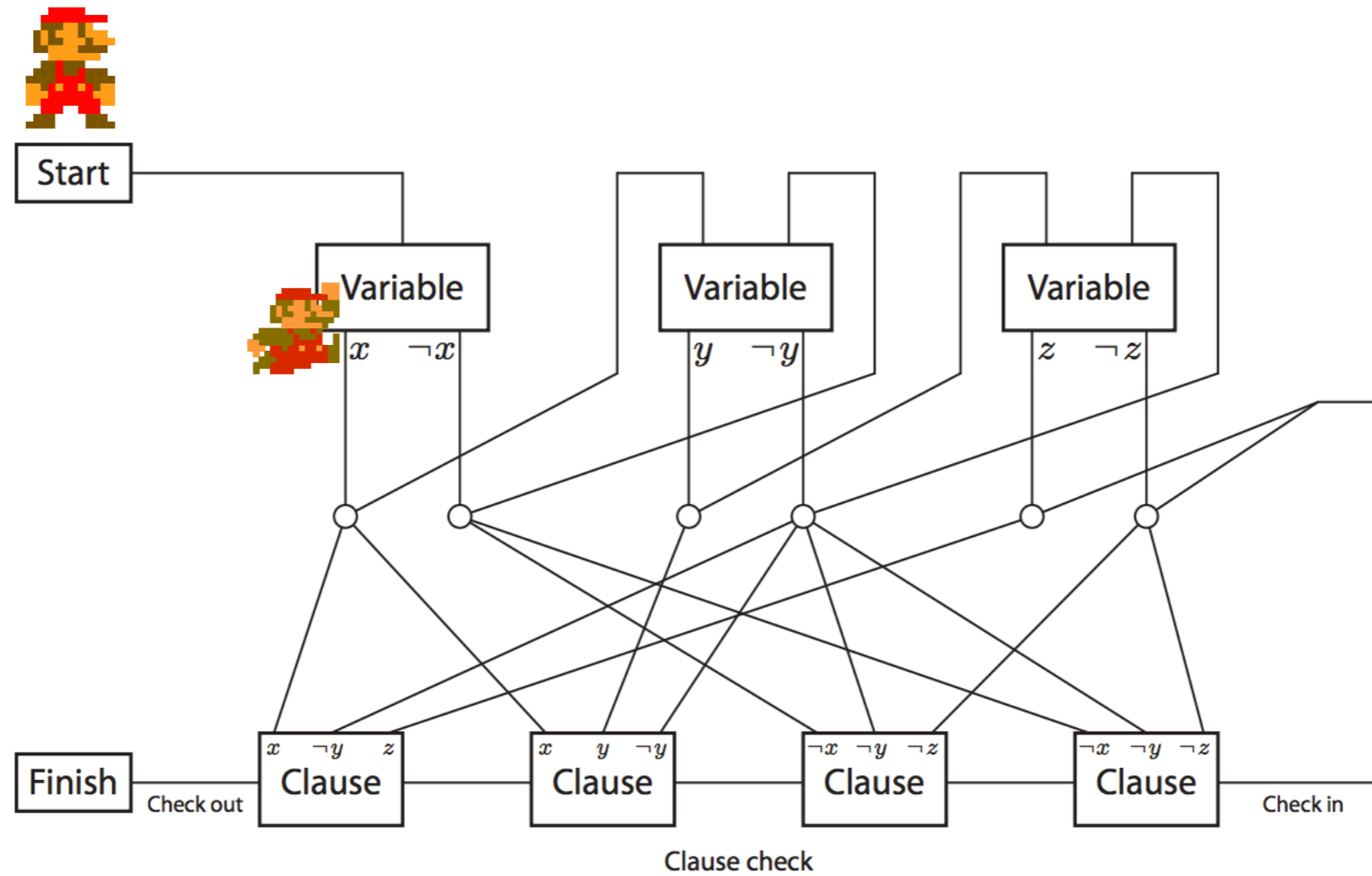
Calculating time complexity when reducing 3-SAT to Mario

- Assume that we have a satisfied 3-SAT problem.
- We can reduce/convert the problem into Mario problem.
- By creating Mario maps out of it.

Creating Mario Maps from 3-SAT

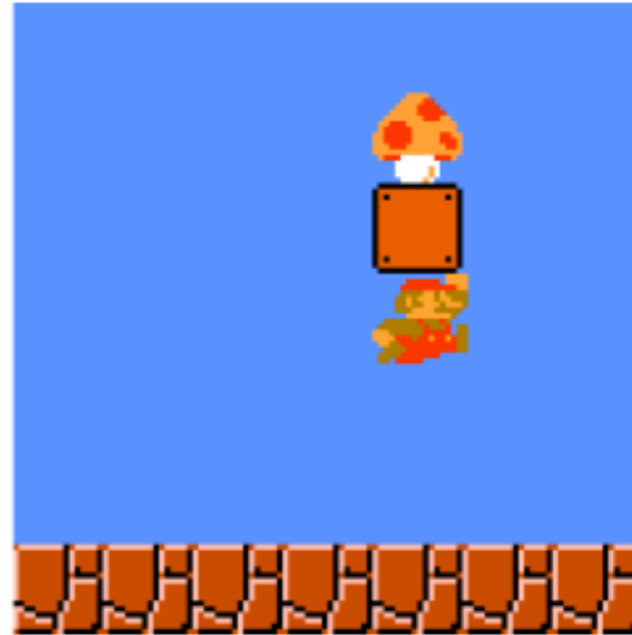
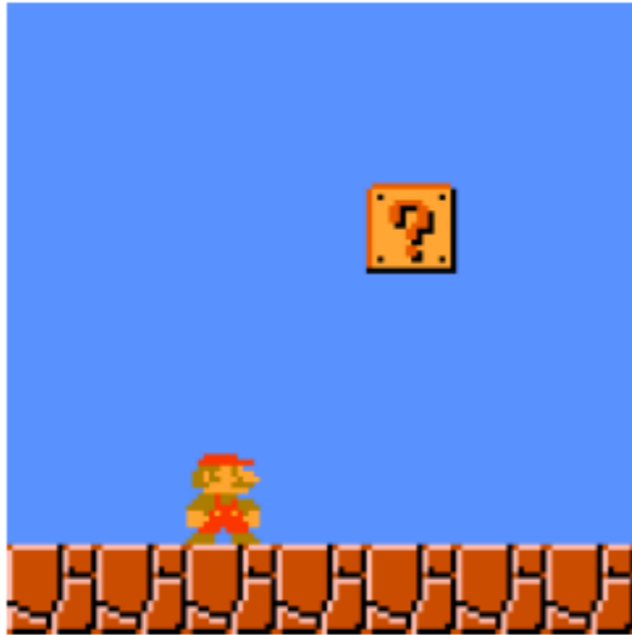
1. Iterate through all clauses $\Rightarrow O(n)$
2. Inside each clause, form a variable gadget for each three literals to form a map $\Rightarrow O(3) \Rightarrow O(1)$
3. Re-iterate the maps (clause gadget) to check validity $\Rightarrow O(n)$
4. Hence, we get $O(n + n + 1) = O(2n + 1) = O(n)$, which means that we reduce 3-SAT to Mario in polynomial time.

Framework

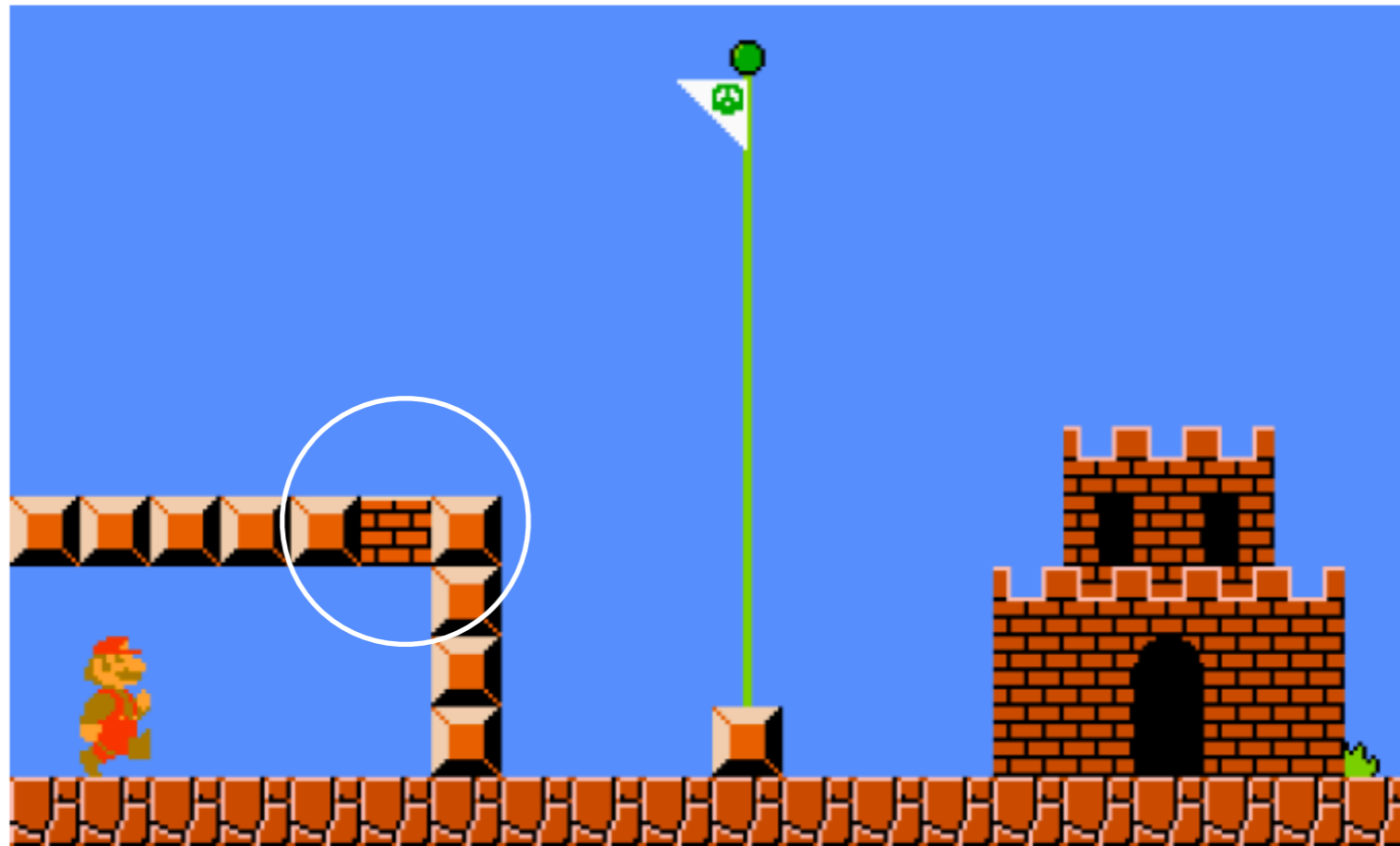


“Assign the variables such that it satisfies the framework”

Start Gadget

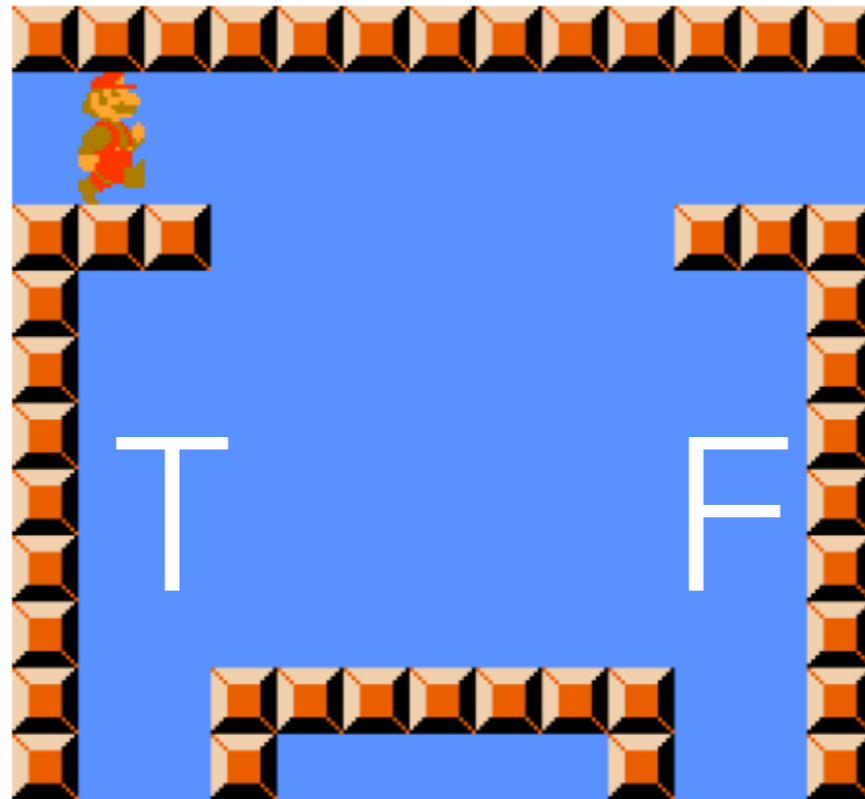


Finish Gadget



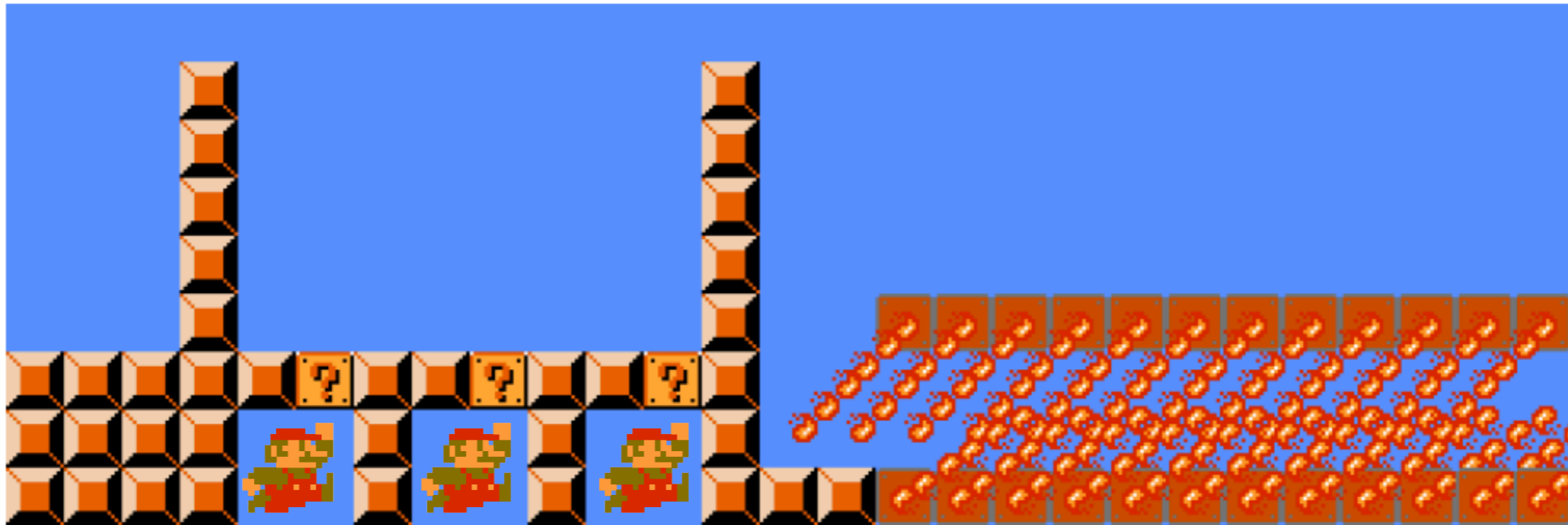
Mushroom is needed!!!

Variable Gadget



X_i

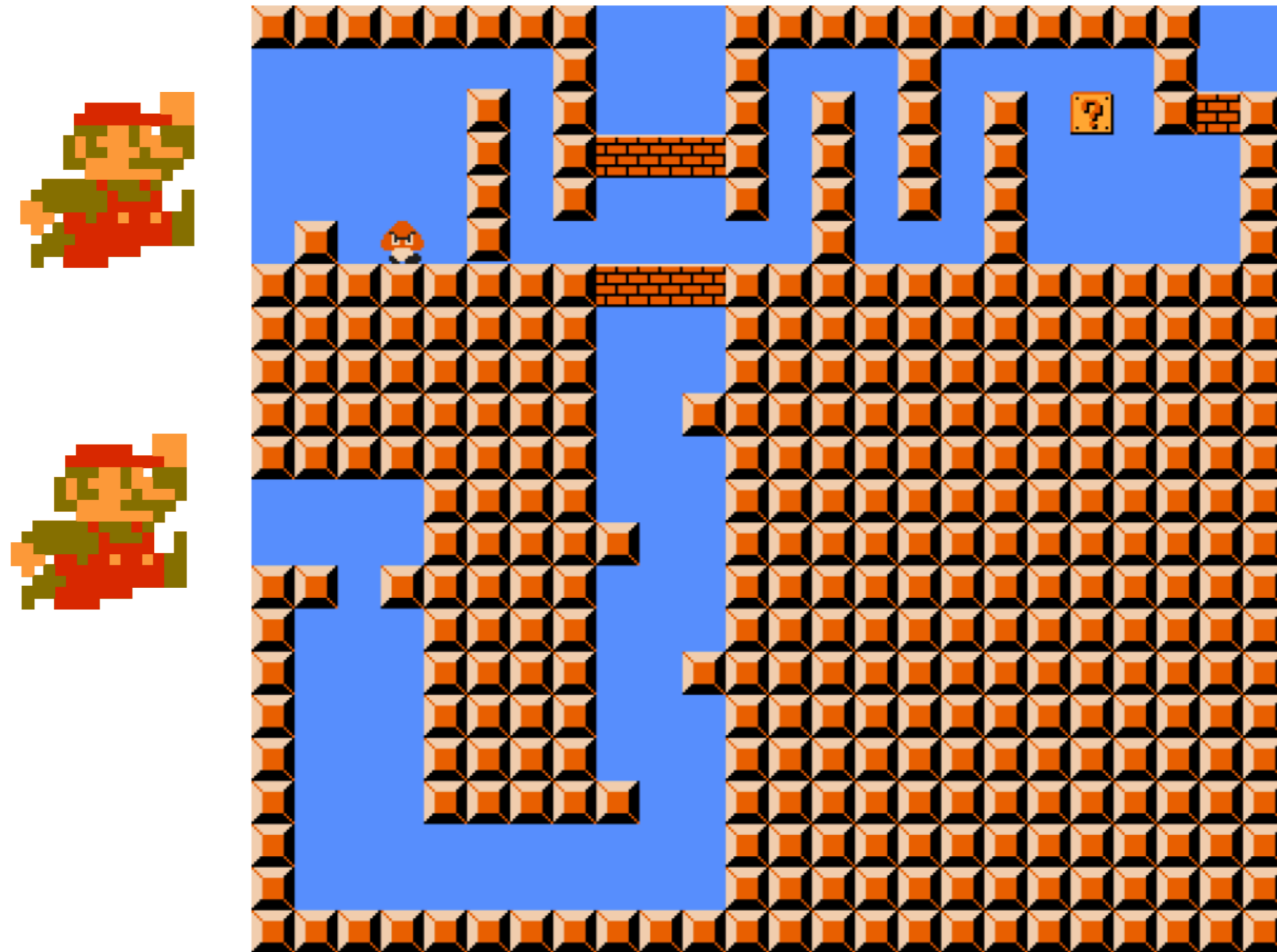
Clause Gadget

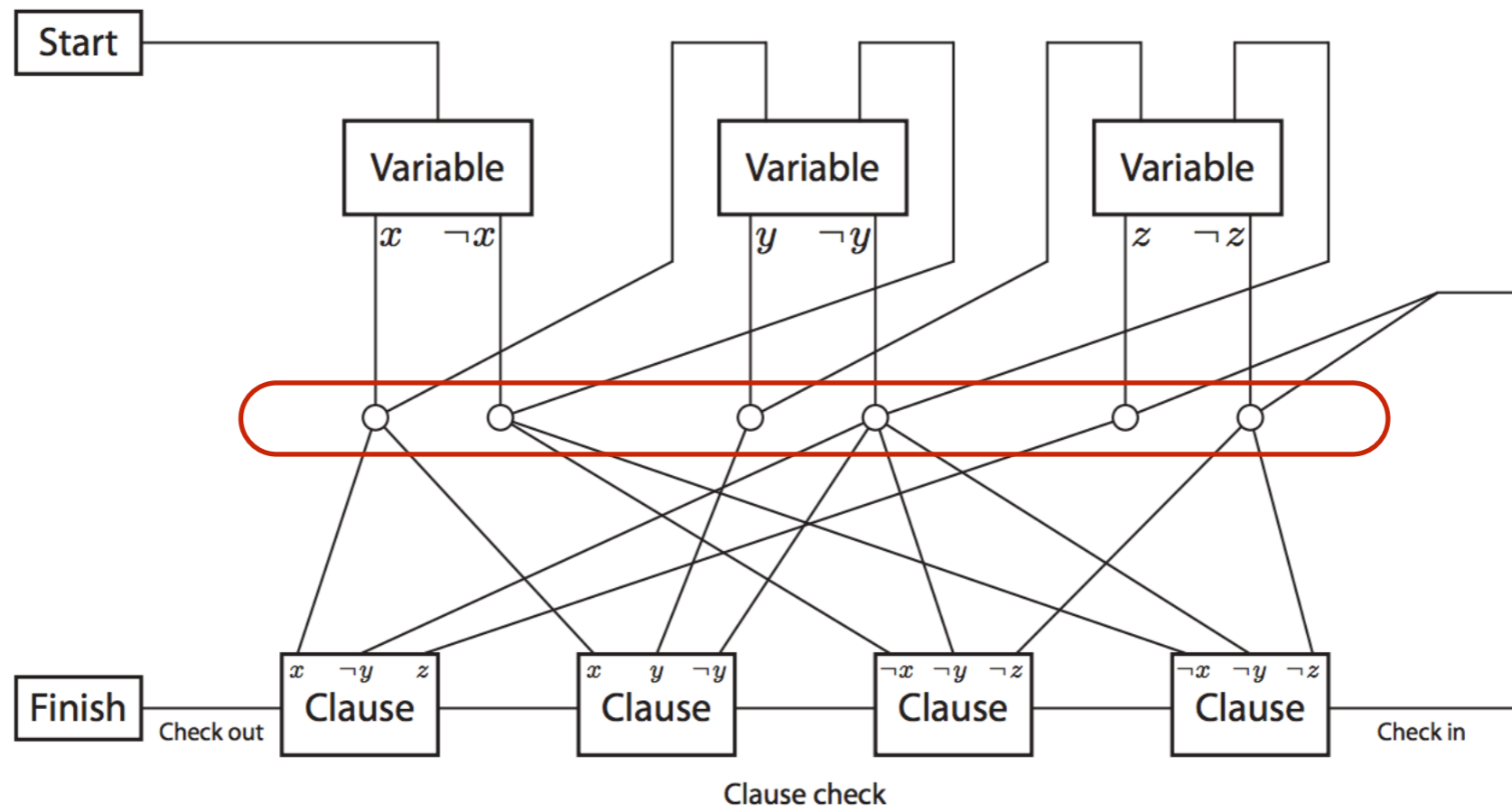


Get the star!!!



Crossover Gadget





“It is NP-Complete to decide whether the goal is reachable from the start of a stage in generalized Super Mario Bros.”