SE2003
Formal Methods in Software Engineering

Spring-2024

Introduction

- Safety critical systems
  - Software controllers
    - Hands free driving
    - Pacemaker
    - Airplane controller software
    - ATM
    - Traffic light controller
- Reliability of software controllers?
  - Does controller satisfy requirements?
    - Software testing
      - Large number of inputs
      - Exhaustive software testing

- Mathematic model
- Formal notations
- Does mathematical model satisfy formal notation?
  - Automated tools
  - Model checking
    - Turing award

Introduction

- Mathematical Model
  - Extensions of finite sate machines
- Formal notations
  - Specify software requirements
- Course Outline

Book: Principles of Model Checking by Christel Baier and Joost Pieter Katoen

Marks Distribution:

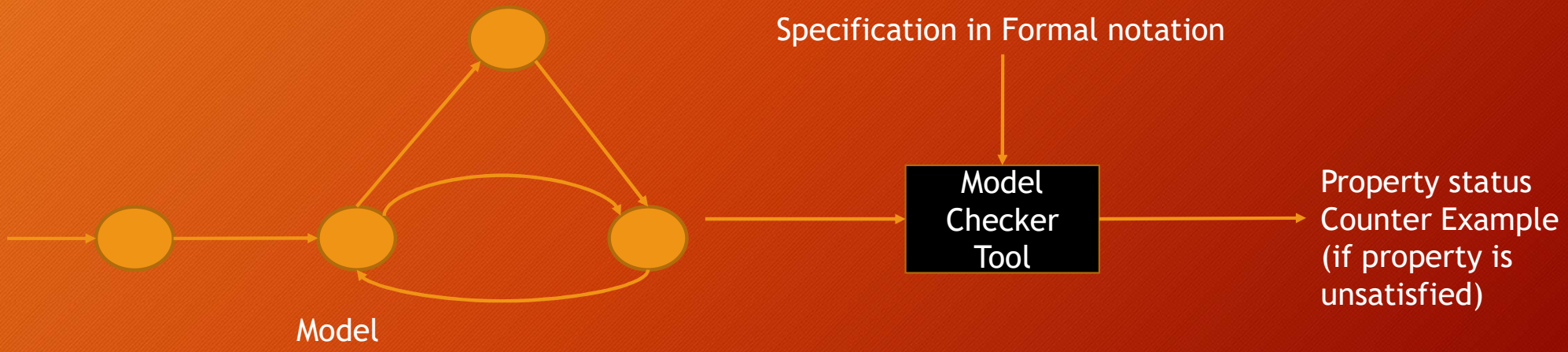
Introduction

- Reliable
  - Decision making is correct?
  - All possible scenarios are considered?
  - Traditional software testing is insufficient.
- Verification and Validation

- Intel's Pentium II processor
  - 1994
  - Error in floating point division code
  - $475 million
- Ariane 5 rocket
  - 1996
  - Crashed in 36 seconds
- Therac 25 radiation therapy machine
  - 1985
  - Death of 6 patients due to radiation overdose

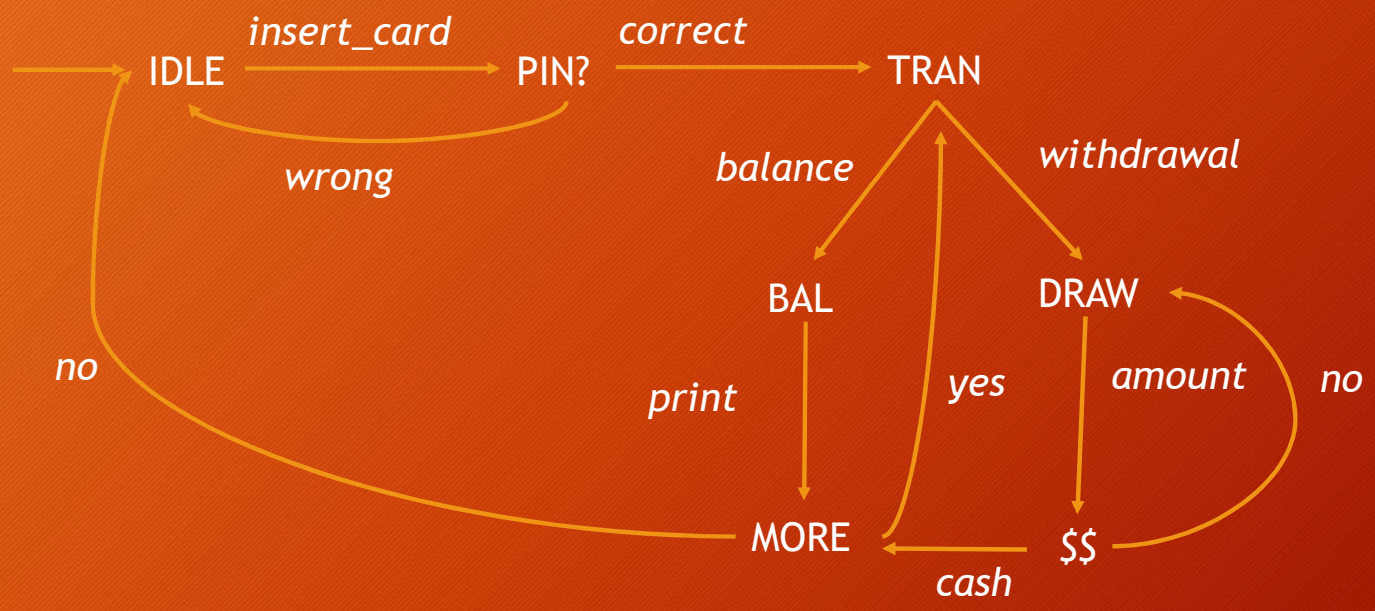
Model Checking

- Finite states machine



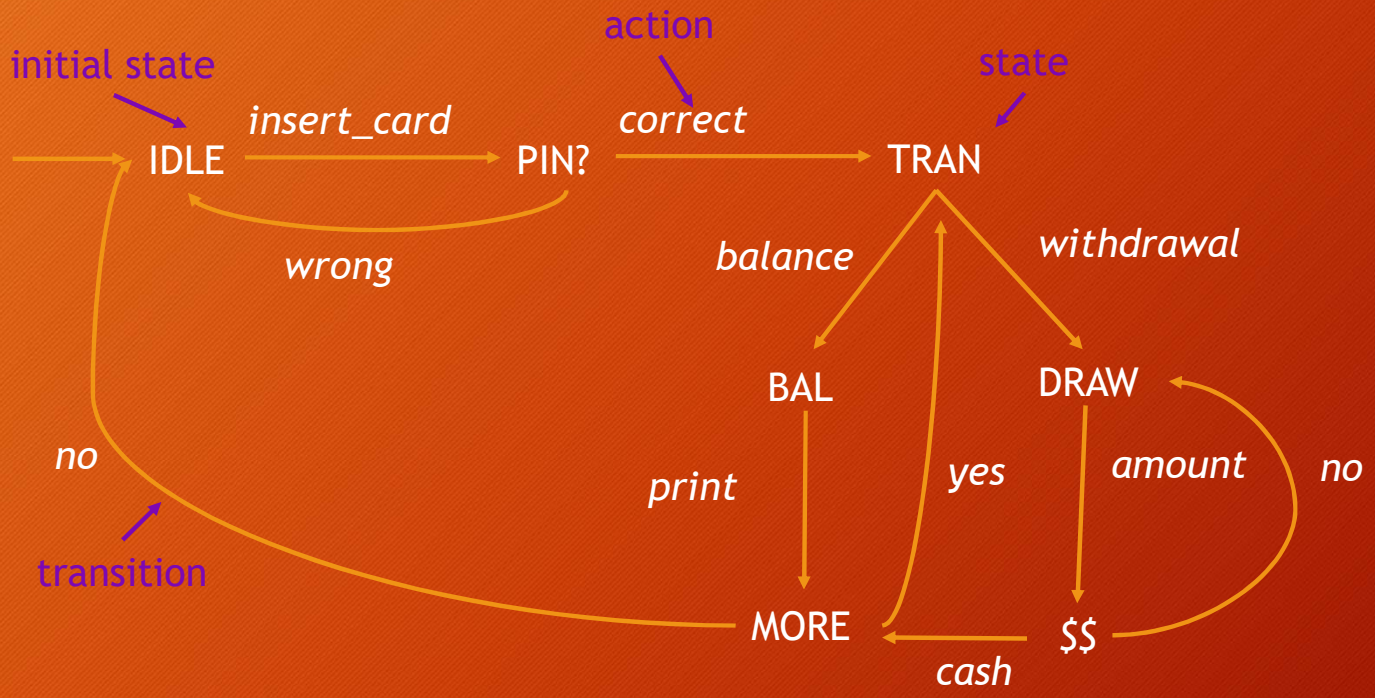
Modeling Code Behavior

- ATM example

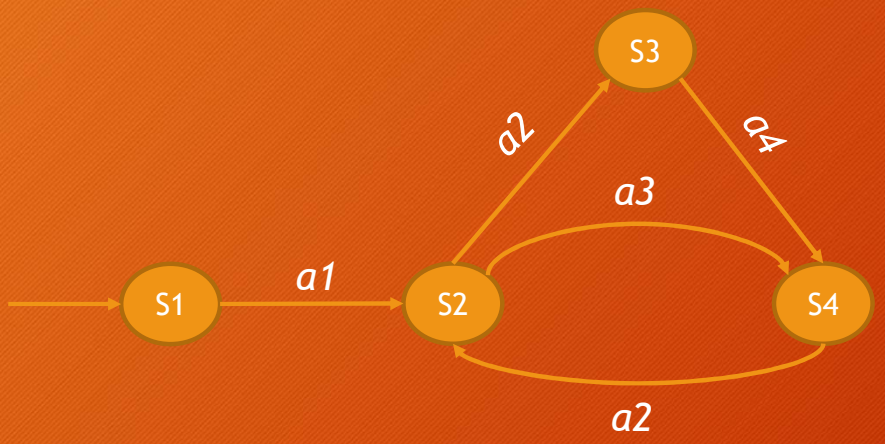


Modeling Code Behavior

- ATM example

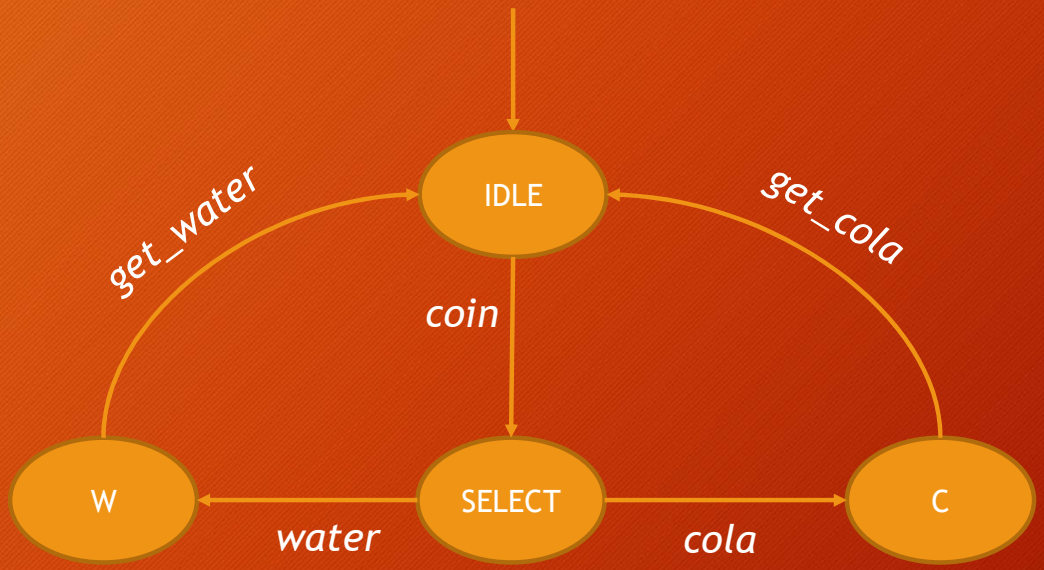


Transition System



- State
- Action
- Transitions
- Initial state

Vending Machine

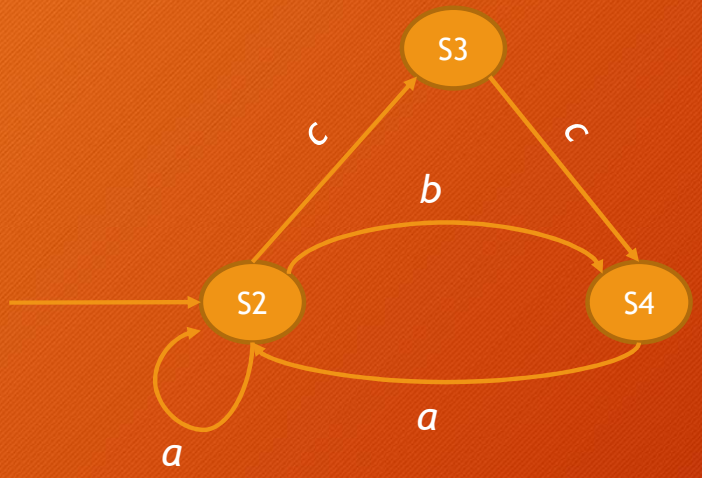


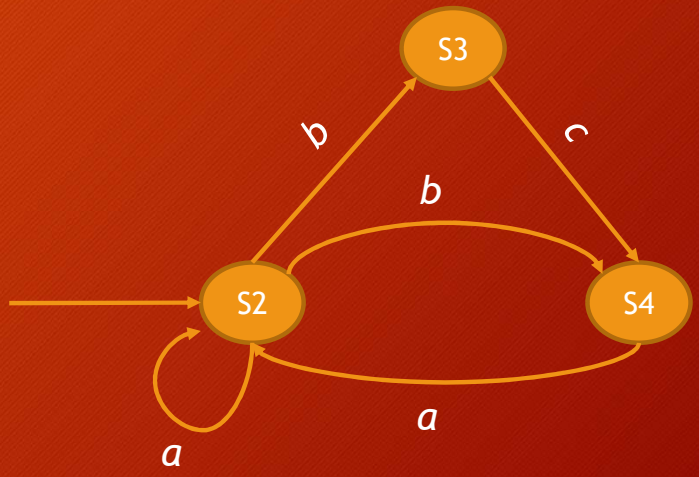
Common Terminology

- Terminal state
- Execution
  - Sequence of transition
  - Represents behavior of a code
    - Finite
      - Ends on a terminal state
    - Infinite
    - Examples

(Non)Deterministic Examples

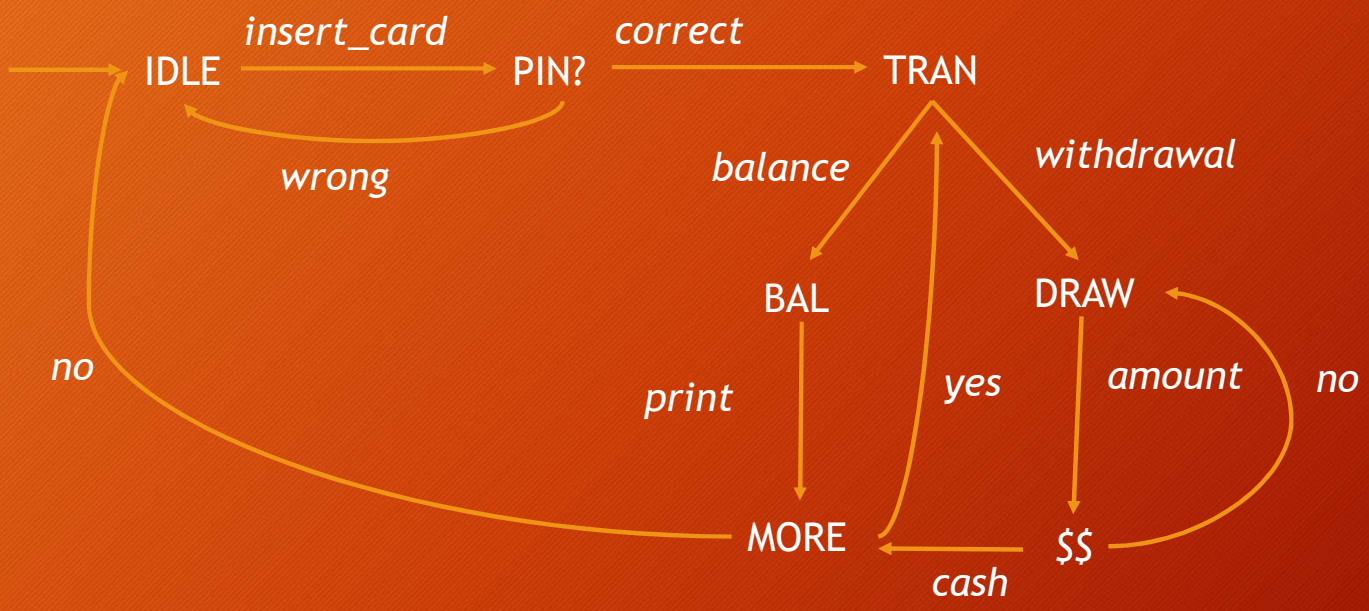
Non-deterministic transition system
- More than one initial state
- More than one transition on an action





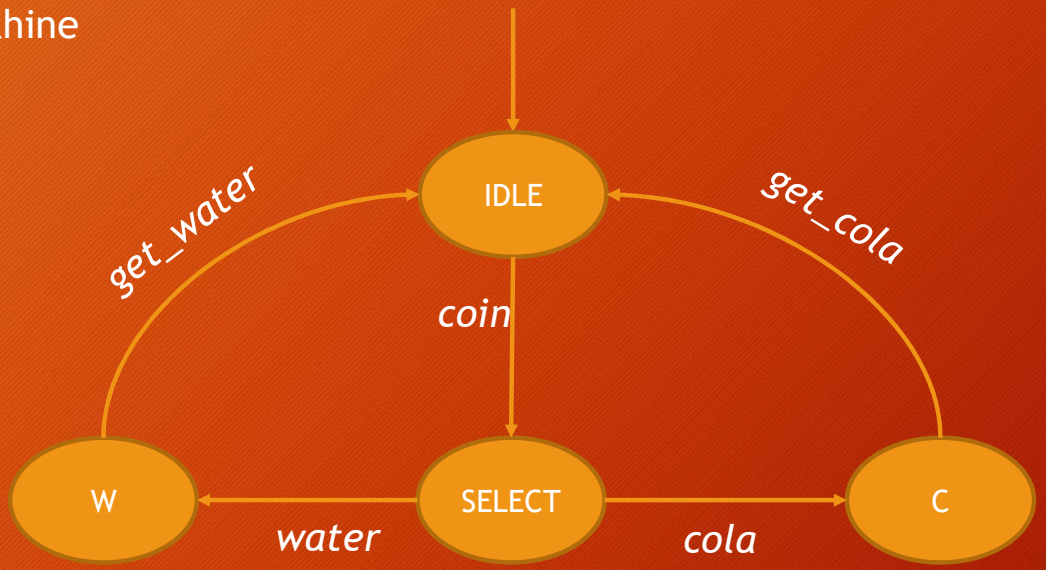
Modeling Code Behavior

- Model of ATM machine



Modeling Code Behavior

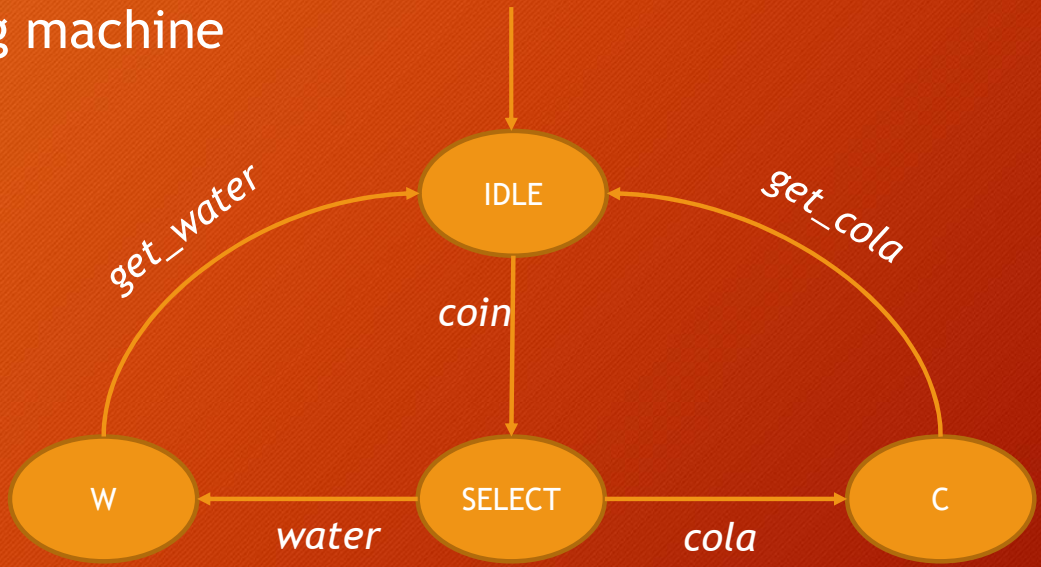
Model of Vending mahine



Modeling data dependent programs

Variables + Conditional branching + Assignments

Modeling vending machine

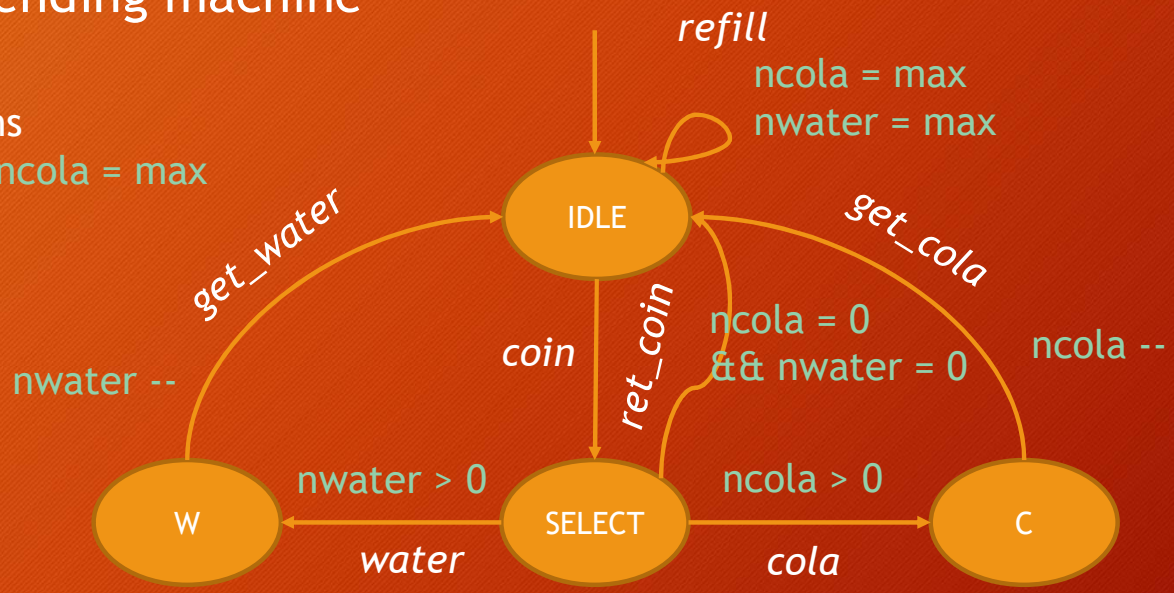


Modeling data dependent programs

Variables + Conditional branching + Assignments

Modeling vending machine

Initial conditions
nwater = max, ncola = max

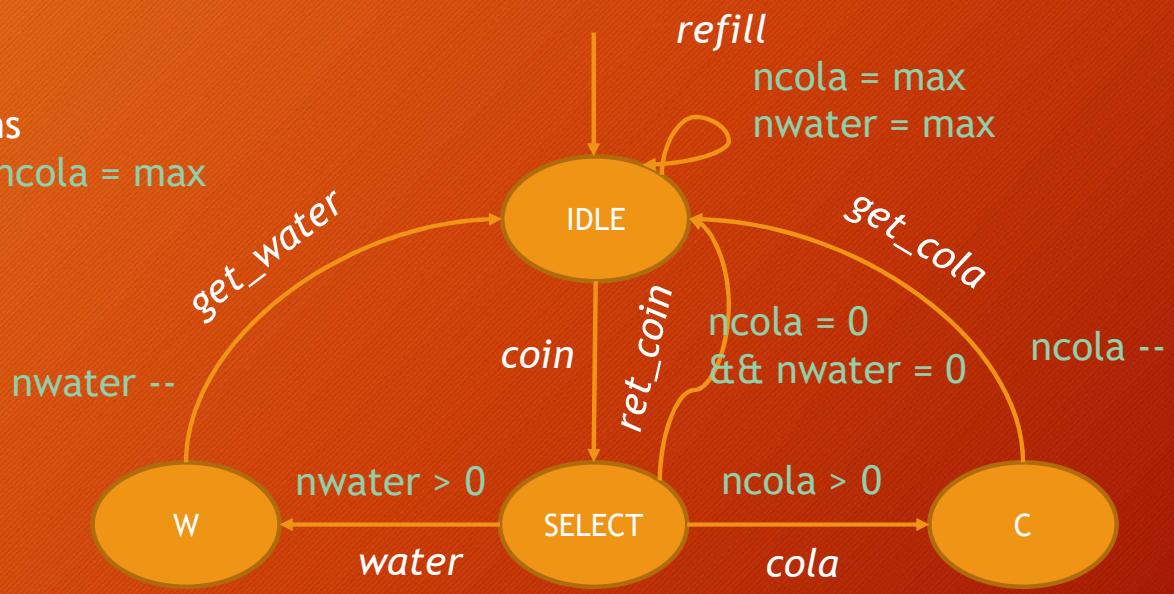


Variables:
nwater, ncola, max

Modeling data dependent programs

Initial conditions
nwater = max, ncola = max

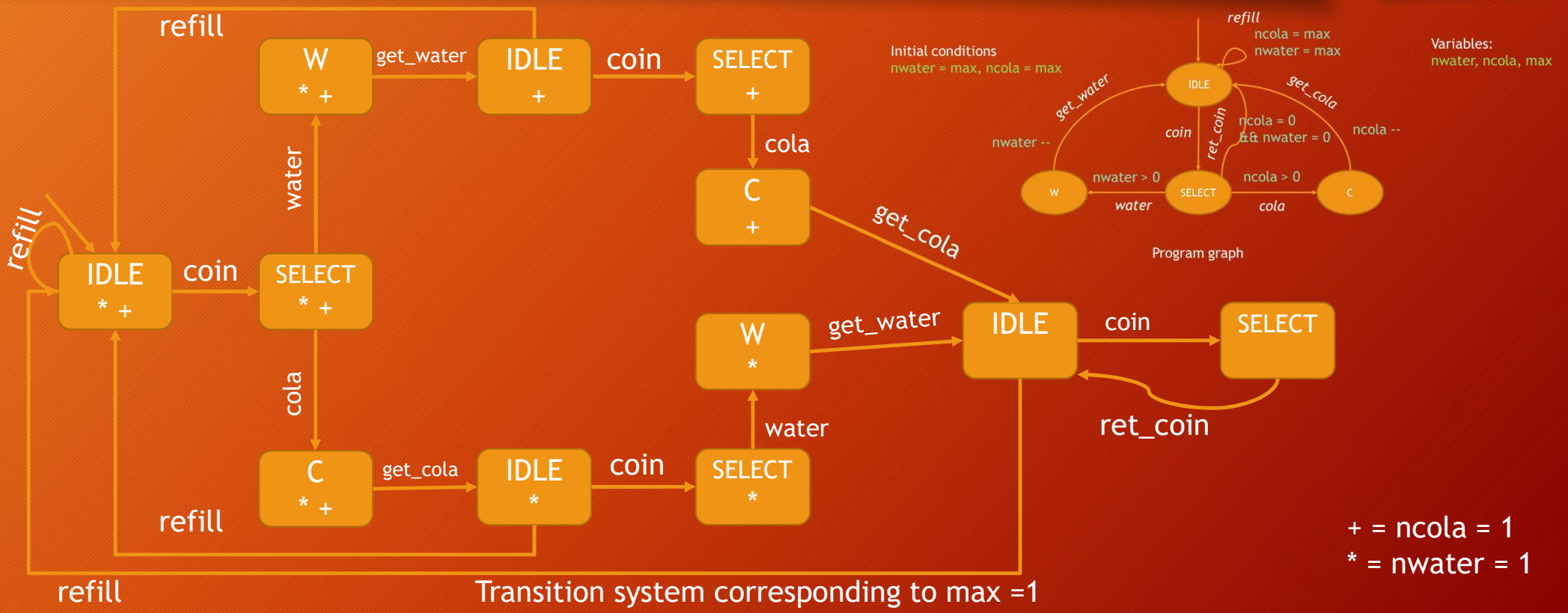
Variables:
nwater, ncola, max



Program graph

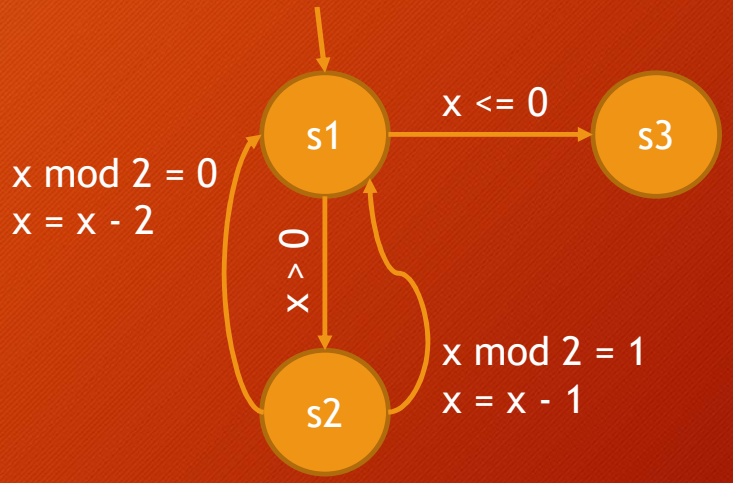
Modeling data dependent programs

max = 1



Modeling data dependent programs

```
...
S1  while (x > 0)
        if (x mod 2 = 0)
                x = x - 2
S2      else
                x = x - 1
S3  ...
```

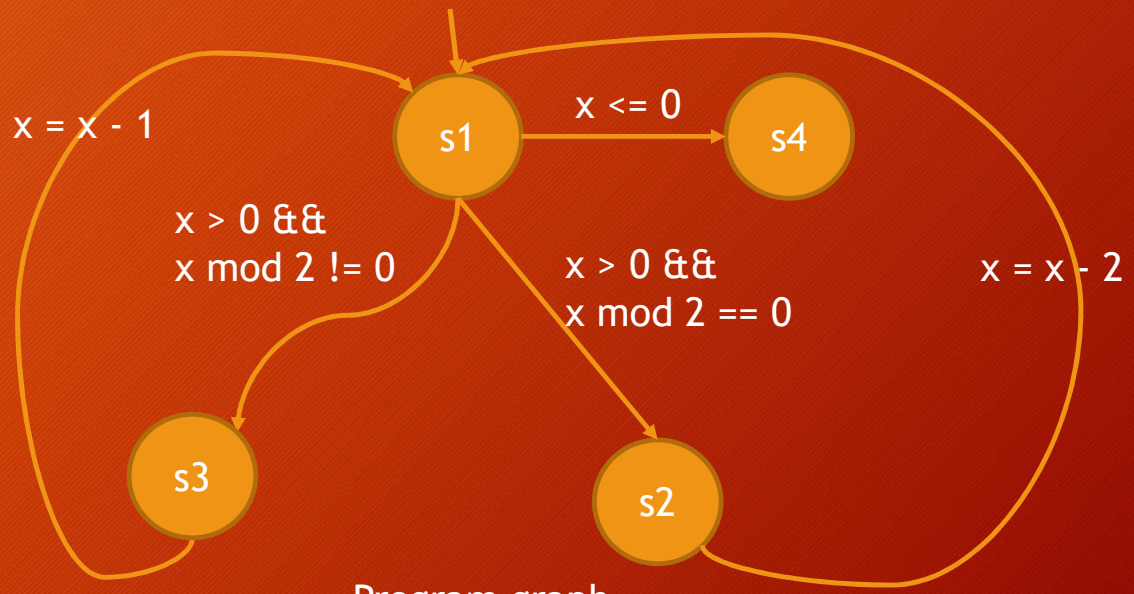


Program graph

Trnsition system with initial condition x = 3

Modeling data dependent programs

```
...
S1  while (x > 0)
        if (x mod 2 = 0)
S2          x = x - 2
        else
S3          x = x - 1
S4  ...
```



Program graph

Transition system with initial value x=5