Class 14: Invariant Principle

Schedule

Problem Set 6 (will be posted after class today) is due **20 October (Friday) at 6:29pm**.

Exam 1 was returned Tuesday. If you did not pick yours up yet, you can get it after class today. We will start charging exponentially-increasing storage fees for inexcusably unclaimed exams starting after Prof. Mahmoody's office hours Monday.

State Machines (review from Class 13)

A *state machine*, $M = (S, G : S \times S, q_0 \in S)$, is a binary relation (called a *transition relation*) on a set (both the domain and codomain are the same set). One state, denoted q_0, is designated as the *start state*.

An *execution* of a state machine $M = (S, G \subseteq S \times S, q_0 \in S)$ is a (possibly infinite) sequence of states, $(x_0, x_1, \cdots, x_n)$ where (1) $x_0 = q_0$ (it begins with the start state), and (2) $\forall i \in \{0, 1, \ldots, n-1\}. (x_i, x_{i+1}) \in G$ (if q and r are consecutive states in the sequence, then there is an edge $q \rightarrow r$ in G).

A state q is *reachable* if it appears in some execution.

A *preserved invariant* of a state machine $M = (S, G \subseteq S \times S, q_0 \in S)$ is a predicate, P, on states, such that whenever $P(q)$ is true of a state q, and $q \rightarrow r \in G$, then $P(r)$ is true.

Bishop State Machine

$S = \{(\_\_\_\_) \,|\, r, c \in \mathbb{N}\}$ $G = \{(r, c) \rightarrow (r', c') \,|\, r, c \in \mathbb{N} \wedge (\exists d \in \mathbb{N}^+ \text{ such that } r' = r\_\_\_\_d \wedge r' \geq 0 \wedge c' = c\_\_\_\_d \wedge c' \geq 0\}$ $q_0 = (0, 2)$

What states are *reachable*?

"Progress" Machine

$S = \{(x, d) \,|\, x \in \mathbb{Z}, d \in \{\mathbf{F}, \mathbf{B}\}\}$ $G = \{(x, \mathbf{F}) \rightarrow (x + 1, \mathbf{B}) \,|\, x \in \mathbb{Z}\} \cup \{(x, \mathbf{B}) \rightarrow (x - 2, \mathbf{F}) \,|\, x \in \mathbb{Z}\}$ $q_0 = (0, \mathbf{F})$

Which states are *reachable*?

Preserved Invariants

A predicate $P(q)$ is a *preserved invariant* of machine $M = (S, G \subseteq S \times S, q_0 \in S)$ if:

$$\forall q \in S.\, (P(q) \land (q \to r) \in G) \implies P(r)$$

What are some *preserved invariants* for the (original) Bishop State Machine?

Invariant Principle. If a *preserved invariant* of a state machine is true for the start state, it is true for all reachable states.

To show $P(q)$ for machine $M = (S, G \subseteq S \times S, q_0 \in S)$ all $q \in S$, show:

1. Base case: $P(\_\_\_\_)$
2. $\forall s \in S.\, \_\_\_\_ \implies \_\_\_\_$

Prove that the original Bishop State Machine never reaches a square where $r + c$ is odd.

Slow Exponentiation

```python
def slow_power(a, b):
   y = 1
   z = b
   while z > 0:
      y = y * a
      z = z - 1
   return y
```

$S ::= \mathbb{N} \times \mathbb{N}$ $G ::= \{(y, z) \rightarrow (y \cdot a, z - 1) \,|\, \forall y, z \in \mathbb{N}^{+}\}$ $q_0 ::= (1, b)$

Prove `slow_power(a, b)` = a^b.

Fast Exponentiation

This is the algorithm from Section 6.3.1 written as Python code:

```python
def power(a, b):
   x = a
   y = 1
   z = b
   while z > 0:
      r = z % 2 # remainder of z / 2
      z = z // 2 # quotient of z / 2
      if r == 1:
         y = x * y
      x = x * x
   return y
```

