Lecture 8 - February 2

Math Review

Injection vs. Surjection vs. Bijection
Formulating Arrays
Lab1 Solution Highlights

Injective Functions

no witness to prove violation of inj. prop.

$f \in S \leftrightarrow T$

functional property

$(s, t_1) \in f \wedge (s, t_2) \in f \Rightarrow t_1 = t_2$

rel, fun, partial fun, not total, inj!

$isInjective(f)$
$\iff$
$\forall s_1, s_2, t \bullet (s_1 \in S \wedge s_2 \in S \wedge t \in T) \Rightarrow ((s_1, t) \in f \wedge (s_2, t) \in f \Rightarrow s_1 = s_2)$

to violate injective prop.
$s_1 \neq s_2$ and
s_1, s_2 both map to t.

If f is a ***partial injection***, we write: $\boxed{f \in S \rightarrowtail\!\!\!\!\!\!+\; T}$

- e.g., $\{\varnothing, \{(1, \mathbf{a})\}, \{(2, \mathbf{a}), (3, \mathbf{b})\}\} \subseteq \{1, 2, 3\} \rightarrowtail\!\!\!\!\!\!+\; \{a, b\}$
- e.g., $\{(1, \mathbf{b}), (2, a), (3, \mathbf{b})\} \notin \{1, 2, 3\} \rightarrowtail\!\!\!\!\!\!+\; \{a, b\}$
- e.g., $\{(1, \mathbf{b}), (3, \mathbf{b})\} \notin \{1, 2, 3\} \rightarrowtail\!\!\!\!\!\!+\; \{a, b\}$

the set of all possible partial injections between two sets

If f is a ***total* injection***, we write: $\boxed{f \in S \rightarrowtail T}$

all possible total inj.

- e.g., $\boxed{\{1, 2, 3\} \rightarrowtail \{a, b\}} = \varnothing$ $\{(1, a), (2, b), (3, a)\}$
- e.g., $\{(2, d), (1, a), (3, c)\} \in \{1, 2, 3\} \rightarrowtail \{a, b, c, d\}$
- e.g., $\{(2, d), (1, c)\} \notin \{1, 2, 3\} \rightarrowtail \{a, b, c, d\}$ not total, inj.
- e.g., $\{(2, \mathbf{d}), (1, c), (3, \mathbf{d})\} \notin \{1, 2, 3\} \rightarrowtail \{a, b, c, d\}$

total, not inj.

rel, partial fun, total fun, t

$\overset{s_1}{(1, b)} \in f \wedge \overset{s_2}{(3, b)} \in f \Rightarrow \overset{F}{\boxed{1 = 3}}$ violation

Surjective Functions

assumed: f is a function.

$$\texttt{isSurjective}(f) \iff \underline{\mathrm{ran}}(f) = \underline{T}$$

If f is a ***partial surjection***, we write: $\boxed{f \in S \twoheadrightarrow\!\!\!\!\!\!+\; T}$

rel, partial, total, sur.

- e.g., $\{\{(1,\mathbf{b}),(2,\mathbf{a})\},\{(1,\mathbf{b}),(2,\mathbf{a}),(3,\mathbf{b})\}\} \subseteq \{1,2,3\} \twoheadrightarrow\!\!\!\!\!\!+\; \{a,b\}$
- e.g., $\{(2,\mathbf{a}),(1,\mathbf{a}),(3,\mathbf{a})\} \notin \{1,2,3\} \twoheadrightarrow\!\!\!\!\!\!+\; \{a,b\}$
- e.g., $\{(2,\mathbf{b}),(1,\mathbf{b})\} \notin \{1,2,3\} \twoheadrightarrow\!\!\!\!\!\!+\; \{a,b\}$

rel, part. fun. sur. not total

If f is a ***total surjection***, we write: $\boxed{f \in S \twoheadrightarrow T}$

- e.g., $\{\boxed{\{(2,a),(1,b),(3,a)\},\{(2,b),(1,a),(3,b)\}}\} \subseteq \{1,2,3\} \twoheadrightarrow \{a,b\}$
- e.g., $\{(2,a),(3,b)\} \notin \{1,2,3\} \twoheadrightarrow \{a,b\}$ surj. $\underline{\underline{\text{not}}}$ total
- e.g., $\underline{\{(2,\mathbf{a}),(3,\mathbf{a}),(1,\mathbf{a})\}} \notin \{1,2,3\} \twoheadrightarrow \{a,b\}$

total, $\underline{\underline{\text{not}}}$ sur.

$$\boxed{\begin{array}{l} \text{partial} \not\Rightarrow \text{total} \\ \text{total} \Rightarrow \text{partial} \end{array}}$$

total surjections ✓ $\Rightarrow$ partial surjections.

If f is a ***total surjection***, we write: $\boxed{f \in S \twoheadrightarrow T}$ X

- e.g., $\{\{(2, a), (1, b), (3, a)\}, \boxed{\{(2, \cancel{b}), (1, a), (3, \cancel{b})\}}\} \subseteq \{1, 2, 3\} \twoheadrightarrow \{a, b\}$ F

a a

$\notin \{1, 2, 3\} \twoheadrightarrow \{a, b\}$

Bijective Functions

f is ***bijective***/***a bijection***/*one-to-one correspondence* if f is ***total***, ***injective***, and ***surjective***.

- e.g., $\boxed{\{1,2,3\} \twoheadrightarrow \{a,b\} = \varnothing}$ → cannot be injective ⇒ cannot be bijective
- e.g., $\{\{(1,a),(2,b),(3,c)\},\{(2,a),(3,b),(1,c)\}\} \subseteq \{1,2,3\} \twoheadrightarrow \{a,b,c\}$
- e.g., $\{(\mathbf{2},b),(\mathbf{3},c),(\mathbf{4},a)\} \notin \{1,2,3,4\} \twoheadrightarrow \{a,b,c\}$
- e.g., $\{(1,\mathbf{a}),(2,b),(3,c),(4,\mathbf{a})\} \notin \{1,2,3,4\} \twoheadrightarrow \{a,b,c\}$
- e.g., $\{(1,\mathbf{a}),(2,\mathbf{c})\} \notin \{1,2\} \twoheadrightarrow \{a,b,c\}$

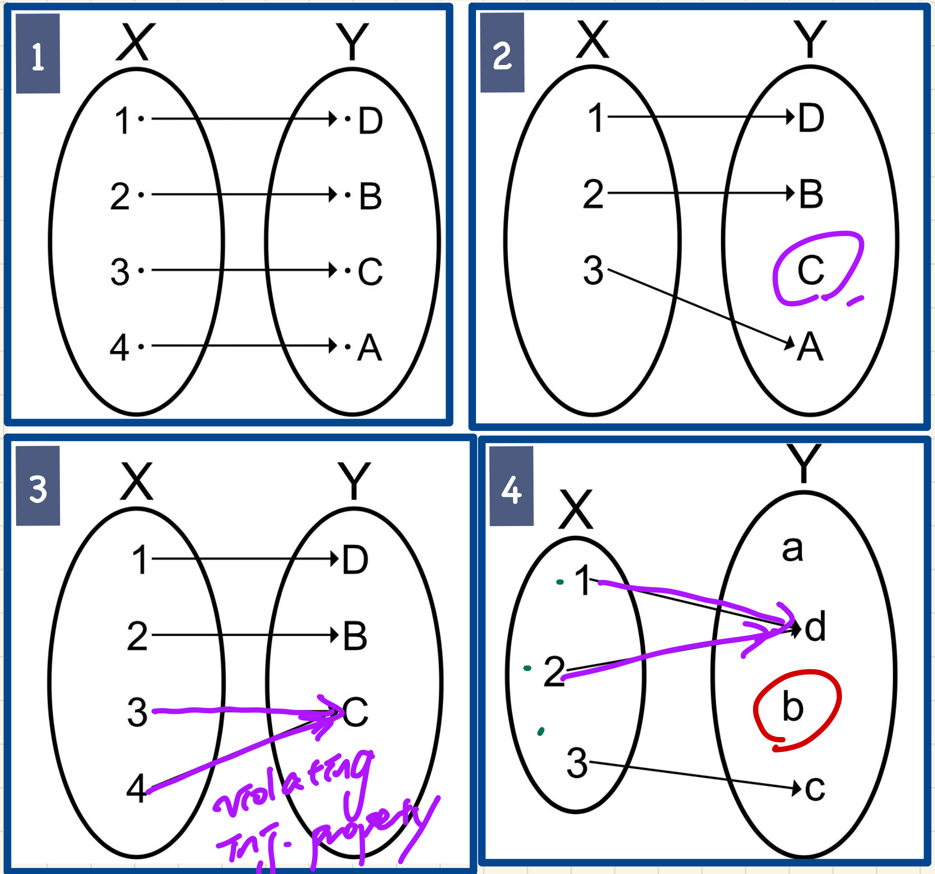
not total

inj,
not sur,
total

not inj.
sur.
total

Exercise

$\in X \leftrightarrow Y$



	1	2	3	4
partial	✓	✓	✓	✓
total	✓	✓	✓	✓
injection	✓	✓	✗	✗
surjection	✓	✗	✓	✗
bijection	✓	✗	✗	✗

Formalizing Arrays as Functions

String[] names = {"alan", "mark", "tom"};

0	1	2
"alan"	"mark"	"tom"

names

$names \in \mathbb{Z} \rightarrow String$

not appropriate

$\{(0, \text{"alan"}), (1, \text{"mark"}), (2, \text{"tom"})\}$

$$\boxed{names \in \mathbb{Z} \leftrightarrow String}$$

No!

(1) e.g. $\{(0, \text{"alan"}), (0, \text{"jim"})\}$

an index should only hold <u>at most one</u> value

(2) e.g. $\{(-1, \text{"jonathan"}), (3, \text{"peter"})\}$

invalid indices

a

0	1	2
"alan"	"mark"	"alan"

$\{(0, \text{"alan"}), (1, \text{"mark"}), (2, \text{"alan"})\}$

not inj ⇒ duplicates

$a \in \mathbb{Z} \mathrel{+\!\!\!\!\twoheadrightarrow} \boxed{String}$

length: 0

1

2 "ab" "bc" . . .

⋮

Array

$$a \in \mathbb{Z} \nrightarrow \text{Object}$$

$$\mathbb{N}$$

$$a \in \mathbb{N} \to \text{Object}$$

not good ∵ indices may be too large.

CONTEXT C0

SETS

ACCOUNT carrier set: abstract without the need to enumerate content of the set

PERSON carrier set: details of each member in PERSON are abstracted away (ENV9) - Solution to Exercise 4 of Lab1

CONSTANTS

c credit limit (ENV3)

L pre-set upper bound (ENV3) - Solution to Exercise 3 of Lab1

AXIOMS

`axm1`: $c \in \mathbb{N}_1$

not theorem means an axiom; theorem means a proof is needed. In this case, the typing constraint should be an axiom.

`thm1`: ⟨theorem⟩ $c > 0$

`axm2`: $L \in \mathbb{N}_1$

typing constraint of variable L - Solution to Exercise 3 of Lab1

END

MACHINE Bank0
  // Initial model of the bank system
SEES C0
VARIABLES
  b balance (ENV2)
  d cash drawer (REQ7)
  owner account owner (ENV9) - Solution to Exercise 4 of Lab1
INVARIANTS
  inv1: $b \in ACCOUNT \rightarrow\!\!\!\!\!+ \mathbb{Z}$
  inv2: $d \in \mathbb{Z}$
  $\boxed{\text{inv3}}$: $\forall a \cdot a \in dom(b) \Rightarrow b(a) \geq -c$
    (ENV3)

Cannot be satisfied simultaneously (e.g. when every account has -c balance)

  inv4: $\forall a \cdot a \in dom(b) \Rightarrow b(a) \leq L$
    (ENV3) - Solution to Exercise 3 of Lab1
  inv5: $owner \in ACCOUNT \rightarrow\!\!\!\!\!+ PERSON$
    (ENV9) - Solution to Exercise 4 of Lab1
  inv6: $dom(b) = dom(owner)$
    Consistent domains of the balance and owner functions (ENV9) - Solution to Exercise 4 of Lab1 (Note. If we declared this invariant as a theorem, then it must be provable/derivable from other invariants that are declared as axioms, which is not the case. Instead, we also declare this invariant as an axiom (i.e., not as a theorem) so that proof obligations (POs) will be generated regarding it being established (by INITIALIZATION) and preserved (by other events).)
  $\boxed{\text{inv7}}$: $d > 0$

Contradicts with

    $\boxed{\text{REQ8}}$ - this was not assigned as a tak for your Lab1. But encoding REQ8 as an invariant shows the value of a formal tool like Rodin: information requirements like E- and R-descriptions are likely to cotain contradictions which are not easy to detect.
EVENTS
Initialisation
  begin
    act1: $b := \varnothing$
    act2:
      $d := 0$

      (REQ4)
    act3: $owner := \varnothing$
      Empty bank (ENV9) - Solution to Exercise 4 of Lab1
  end
Event withdraw ⟨ordinary⟩ $\widehat{=}$
  (REQ6) - Exercise 2 from Lab1: withdraw/inv3/INV cannot be proved.
  any
    a account to withdraw
    v value to withdraw
  where
    type_of_a: $a \in ACCOUNT$
      typing constraint of event parameter a
    type_of_v: $v \in \mathbb{N}_1$
      typing constraint of event parameter v
    wd_for_b(a): $a \in dom(b)$
    inv_3: $b(a) - v \geq -c$
      Solution to Exercise 2 of Lab1
  then
    act1: $b(a) := b(a) - v$
      updates the balance of a
    act2: $d := d - v$
      updates the cash drawer
  end

Event deposit ⟨ordinary⟩ $\widehat{=}$
(REQ5) - Solution to Exercise 3 of Lab1
any
a
v
where
grd1: $a \in dom(b)$
grd2: $v \in \mathbb{N}_1$
grd3: $b(a) + v \leq L$
then
act1: $b(a) := b(a) + v$
act2: $d := d + v$
end

Event open_account ⟨ordinary⟩ $\widehat{=}$
(REQ4) - Solution to Exercise 4 of Lab1
any
p
a
where
grd1: $p \in PERSON$
grd2: $a \in ACCOUNT$
grd3: $a \notin dom(owner)$
then
act1: $b := b \cup \{a \mapsto 0\}$
Note. Might need the PP prover to discharge POs related to inv3/inv4
act2: $owner := owner \cup \{a \mapsto p\}$
end

Event close_account ⟨ordinary⟩ $\widehat{=}$
(REQ10) - Solution to Exercise 4 of Lab1
any
a
where
grd1: $a \in dom(b)$
grd2: $b(a) = 0$
then
act1: $b := \{a\} \lhd\!\!\!- b$
act2: $owner := \{a\} \lhd\!\!\!- owner$
end

Event transfer ⟨ordinary⟩ $\widehat{=}$
(REQ11) - Solution to Exercise 4 of Lab1
any
a1
a2
v
where
grd1: $a1 \in dom(b)$
grd2: $a2 \in dom(b)$
grd3: $a1 \neq a2$
grd4: $b(a1) - v \geq -c$ overriding
grd5: $b(a2) + v \leq L$
grd6: $v \in \mathbb{N}_1$
Necessary to make POs related to inv3/inv4 discharged
then
t :
act1: $b := b \Leftarrow \{a1 \mapsto b(a1) - v, a2 \mapsto b(a2) + v\}$
Note. It's not allowed to have two actions involving the same LHS variable: b(a1) := ... , b(a2) := ...
end
END

Rewrite

$b := t \cup \{a1, a2\} \lhd\!\!\!- b$