CS 135
Sets and functions, 1

Professor Bloom

January 23, 2009

1 Functions

Suppose that A, B are sets. A **function** $f : A \to B$ is a rule which assigns to each member $a \in A$ an element $f(a) = (f\ a)$ in B. The set A is **the source of** f and the set B is the **target of** f.

Examples

- If $A = \mathbb{Z}$, the integers, and $B = \mathbb{N}$, the absolute value is a function $\mathbb{Z} \to \mathbb{N}$.
- If $A = B = \mathbb{N}_n = \{0, 1, \ldots, n-1\}$, $f = (\lambda x)(x^2 \mod n)$ is the function, whose value at $x \in A$ is $(x^2 \mod n$. Thus, when $n = 6$, $(f\ 3) = 3$.
- If A is a set of people, `age` is a function $A \to \mathbb{N}$ that assigns to each person in A her age.
- For any set A, there is a function

  $$1_A : A \quad \to \quad A$$

  which assigns to $x \in A$ the element x. This function is called the *identity function on A*.
- By convention, for any set A there is a unique function $\emptyset \to A$.

2 Function Composition

Suppose that $f : A \to B$ and $g : B \to C$ are functions. The **composite** of f, g is the function $h : A \to C$ whose value on $a \in A$ is the value of g on the element

$(f\ a)$.

$$\begin{aligned} h &= A \xrightarrow{f} B \xrightarrow{g} C \\ (h\ a) &= (g\ (f\ a)). \end{aligned}$$

The function h is usually written $g \cdot f$.

Composition has two fundamental properties.

1. If $f : A \to B,\ g : B \to C,\ h : C \to D$ are functions, then

$$h \cdot (g \cdot f) \quad = \quad (h \cdot g) \cdot f.$$

2. If $f : A \to B$, then

$$\begin{aligned} f &= f \cdot 1_A \\ &= 1_B \cdot f. \end{aligned}$$

Examples

1. Suppose $f : \mathbb{N} \to \mathbb{N}$ is the function $(f\ x) = 2x$, and $g : \mathbb{N} \to \mathbb{N}$ is $(g\ x) = 3x + 2$. Then

$$\begin{aligned} (f \cdot g\ x) &= 2(3x + 2) \\ (g \cdot f\ x) &= 3(2x) + 2. \end{aligned}$$

2. Suppose $f : \mathbb{N} \to \mathbb{N}$ is the function $(f\ x) = 0$, and $g : \mathbb{N} \to \mathbb{N}$ is $(gx) = 3$. Both are "constant functions". Then

$$\begin{aligned} f \cdot g &= f \\ g \cdot f &= g. \end{aligned}$$

3 Implementing composition in Scheme

This example is one reason Scheme is so popular.

```
(define (comp f g)
   (lambda ( x )
     (g (f x))))
```

Try it! Here is a sample session.

```
> (define (comp f g) (lambda(x) (g (f x))))
> (define h (comp factorial factorial))
> (factorial 10)
3628800
> (h 3)
720
> (h 5)
668950291344912705758811805409037258675274633313802981029
567135230163355724496298936687416527198498130815763789321
40905525344085894081218598984811143896500059649605212569
60000000000000000000000000000
> (h 6)
2601218943565795100204903227081043611191521875016945785727
5418378508356311569473822406785779581304570826199205758922
4725953664156516205201587379198458774083252910524469038881
1884123764341191951045505346658616243271940197113909845536
7272785370993456298555867193697740700037004307837589974206
7678401696720784628062922903210716166986726054898844551425
7193985499448939594496064045132362140265986193073249369770
4776060676806701764916694030348199618814556251955925669188
3082551494294759653727484562462882423452659778973774089646
6553992435928786212515967483220976029505696699927284670563
7471375330192483135870761254126834158601294475660114554207
4958995256354306828863463108496565068277155299625679084523
5702552186222358130016700834523443236821935793184701956510
7297818043541738905607274280485839959197290217266122912984
2051606757903623233769945396419147517556755769539223380305
6825308599977441675784352815913461340394604901269542028838
3471013637338244845066600933484844407119312925376946573543
3737572477223018153403264717753198453734147867432704845798
378661870325740593892421570969599463055752106320326349320
9220738320923356309923267504401701760572026010829288042335
6066430898887102973807975780130560495763428386830571906622
0529117482251053669775660302957404338798347151855260280533
3866357139101046336419769097397432285994219837046979109956
3033896046758898657957111765666700391567481531159439800436
2539939973120306649060132531130471902889849185620376666916
446879112524919375442584589500031156168297430464114253807
4897281723375955380661719801404677935614793635266265683339
5097600000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000
00000000000
```

The factorial function, usually written $n!$, is defined by

$$n! \quad = \quad \begin{cases} 1 & \text{if } n = 0 \\ n * (n-1)! & \text{if } n > 0. \end{cases}$$

Here is a different way to compute it.

```
(define (fact n)
   (define (aux val count)
    (cond
      [(= count n) val]
      [else (aux (* val (add1 count)) (add1 count))]
     ))
   (aux 1 0))
```

How does one define functions in Scheme? The typical way is by an explicit expression.

```
(define square
   (lambda(x)
     (* x x)))
```

The expression "lambda(x)" should be read: the function whose value at x is ....

```
(define identity
  (lambda(x) x))
```

Also,

```
(define expon
   (lambda(n p) ;compute n^p by induction on p
     (cond
       [(= p 0) 1]
       [else (* n (expon n (sub1 p)))]
      )))
```

A function $f : A \to B$ is **injective** or **one-one**, if for any $a, a' \in A$, if $f(a) = f(a')$ then $a = a'$. In other words, if $a \neq a'$ in A, then $f(a) \neq f(a')$. A function $f : A \to B$ is **surjective**, or **onto**, if, for each $b \in B$ there is some $a \in A$ such that $f(a) = b$.

Exercise

1. Find an example of a function $f : A \to A$ which is injective but not surjective.

2. Find an example of a function $f : A \to A$ which is surjective but not injective.

A remarkable fact about finite sets is this.

Theorem 3.1 *Suppose that A is a finite set and $f : A \to A$ is a function. Then f is injective iff f is surjective.*

We prove this fact by induction on the size of A.

3.1 An aside on Induction

Induction is a method to prove facts about the nonnegative integers $\mathbb{N} = \{0, 1, 2, \ldots\}$. Suppose we want to prove every nonnegative integer has a certain property. The **method of induction is**

1. prove 0 has the property (basis step)
2. assume $n \geq 0$ has the property (induction hypothesis)
3. prove $n + 1$ has the property. (induction step)

After proving these two things, we may conclude every nonnegative integer has the property.

There is a an equivalent version, called strong induction.

The **method of strong induction is**

1. assume $n \geq 0$ and assume if $0 \leq x < n$, then x has the property.
2. using this assumption, prove n has the property.

We may conclude every nonnegative integer has the property.

3.2 Proof

Theorem 3.2 *Suppose that A is a finite set and $f : A \to A$ is a function. Then if f is injective then f is surjective.*

By induction on n, the number of elements in A.

Basis. $n = 0$. Then A is empty, and there is a unique funtion $\emptyset \to \emptyset$, which is both injective an surjective.

Induction assumption. $n \geq 0$ and any function on an n element set is injective iff it is surjective.

Induction step. A has $n+1$ elements, and assume $f : A \to A$. Let $a \in A$. Let $B = A - \{a\}$. So B has n elements. There are two cases.

Case 1. $f(a) = a$. Then, if $x \neq a$, $f(x) \in B$, since f is injective. Thus, the function $g : B \to B$ defined by

$$g(x) \quad := \quad f(x)$$

is also injective. By the induction hypothesis, g is surjective. It now follows that f is surjective, since if y is an arbitrary element in A, either $y = a$ or $y \in B$. If $y = a$, $y = f(a)$. If $y \in B$, then $y = g(x) = f(x)$, for some $x \in B$.

Case 2. $f(a) = b \neq a$.

Case 2a. There is some c such that $f(c) = a$. Define g as the following function $A \to A$:

$$g(x) \quad = \quad \begin{cases} a & \text{if } x = a \\ b & \text{if } x = c \\ f(x) & \text{otherwise.} \end{cases}$$

Then g is injective, since f is, and $g(a) = a$. Thus, by Case 1, g is surjective. But then f is surjective, as is easy to see.

Case 2b. $f(a) \neq a$ and there is no c such that $f(c) = a$. Then, for all $x \in B$, $f(x) \in B$. Thus, the function $g : B \to B$ defined by

$$g(x) \quad = \quad f(x)$$

is injective, and hence surjective. But this is impossible, since if $g(x) = f(a)$, f is not injective. Thus, Case 2b is impossible.

We have proved: any injective function on a finite set is surjective.

Theorem 3.3 *If $f : A \to A$ is a function on a finite set A which is surjective, then f is injective.*

Proof. We use the fact, proved in the homework, that there is a function

$$g : A \quad \to \quad A$$

such that

$$f(g(a)) \quad = \quad a, \quad a \in A.$$

But then g is injective, and by the first theorem, also surjective.

We *claim*:

$$g(f(a)) \quad = \quad a, \quad a \in A.$$

Indeed, suppose that $b = f(a)$. Since g is surjective, $a = g(c)$, for some c. But then $f(a) = f(g(c)) = c$. But $f(a) = b$, so that $b = c$. This shows $g(b) = g(f(a)) = a$. But, again by the homework, this equation implies f is injective. □