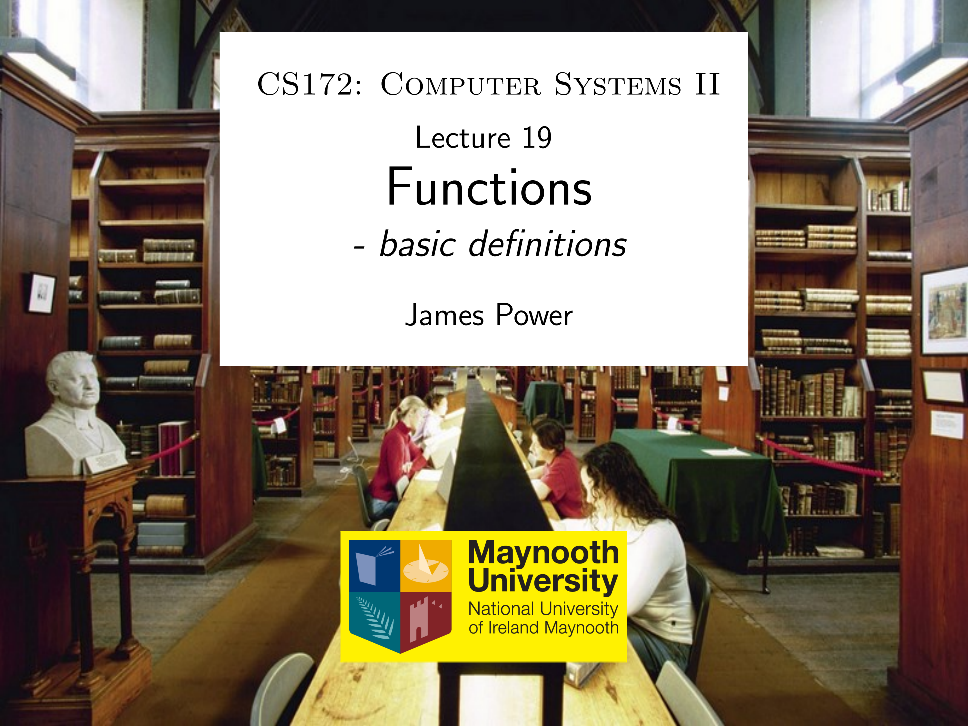


CS172: Computer Systems II

Lecture 19

Functions

- *basic definitions*

James Power

Maynooth University
National University
of Ireland Maynooth

Functions

A function is a special kind of relation.

Functions

A function is a special kind of relation.

- **Definition**: A *binary* relation R is a function if whenever $(a, b) \in R$ and $(a, c) \in R$ then we must have that $b = c$.

Functions

A function is a special kind of relation.

- **Definition**: A *binary* relation R is a function if whenever $(a, b) \in R$ and $(a, c) \in R$ then we must have that $b = c$.
- In general, a relation of arity n is a function if whenever $(a_1, a_2, \ldots a_{n-1}, b) \in R$ and $(a_1, a_2, \ldots a_{n-1}, c) \in R$ then we must have that $b = c$.

Functions

A function is a special kind of relation.

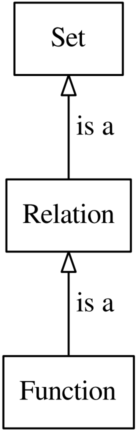
- **Definition**: A *binary* relation R is a function if whenever $(a, b) \in R$ and $(a, c) \in R$ then we must have that $b = c$.
- In general, a relation of arity n is a function if whenever $(a_1, a_2, \ldots a_{n-1}, b) \in R$ and $(a_1, a_2, \ldots a_{n-1}, c) \in R$ then we must have that $b = c$.
- **Notation:** Since a function has this special property, we often use a special notation for being an element of a function.

  For example, if f is a binary relation that is a function, then instead of writing $(a, b) \in f$ we usually write $f(a) = b$

Textbook, §A.5

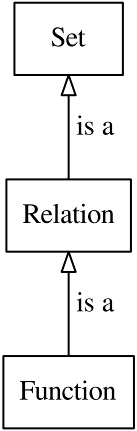
Functions, Relations and Sets (class diagram)

- A relation is a special kind of set.
- A function is a special kind of relation.



Functions, Relations and Sets (class diagram)

- A relation is a special kind of set.
- A function is a special kind of relation.
- Thus all of the *relational operators* (inverse, composition) can work with functions.
- ... and all of the *set-theory operators* (union, intersection etc.) can also work with functions.



Relations vs. functions: numerical examples

Example: consider the following two relations which are both subsets of $\mathbb{Z} \times \mathbb{Z}$:

- $S \triangleq \{a \in \mathbb{Z}, b \in \mathbb{Z} \mid b = a + 1 \bullet (a, b)\}$
- $T \triangleq \{a \in \mathbb{Z}, b \in \mathbb{Z} \mid b = \sqrt{a} \bullet (a, b)\}$

Q1: Is the relation S a function?

Q2: Is the relation T a function?

Relations vs. functions: numerical examples

Example: consider the following two relations which are both subsets of $\mathbb{Z} \times \mathbb{Z}$:

- $S \triangleq \{a \in \mathbb{Z}, b \in \mathbb{Z} \mid b = a + 1 \bullet (a, b)\}$
- $T \triangleq \{a \in \mathbb{Z}, b \in \mathbb{Z} \mid b = \sqrt{a} \bullet (a, b)\}$

Q1: Is the relation S a function? ✓

S *is* a function, since $x + 1$ is uniquely defined for any x.

Q2: Is the relation T a function?

Relations vs. functions: numerical examples

Example: consider the following two relations which are both subsets of $\mathbb{Z} \times \mathbb{Z}$:

- $S \triangleq \{a \in \mathbb{Z}, b \in \mathbb{Z} \mid b = a + 1 \bullet (a, b)\}$
- $T \triangleq \{a \in \mathbb{Z}, b \in \mathbb{Z} \mid b = \sqrt{a} \bullet (a, b)\}$

Q1: Is the relation S a function? ✓

S *is* a function, since $x + 1$ is uniquely defined for any x.

i.e. $\forall\, a \cdot \forall\, b \cdot \forall\, c \cdot ((a, b) \in S \;\wedge\; (a, c) \in S)) \;\rightarrow\; (b = c)$

Q2: Is the relation T a function?

Relations vs. functions: numerical examples

Example: consider the following two relations which are both subsets of $\mathbb{Z} \times \mathbb{Z}$:

- $S \triangleq \{a \in \mathbb{Z}, b \in \mathbb{Z} \mid b = a + 1 \bullet (a, b)\}$
- $T \triangleq \{a \in \mathbb{Z}, b \in \mathbb{Z} \mid b = \sqrt{a} \bullet (a, b)\}$

Q1: Is the relation S a function? ✓

S *is* a function, since $x + 1$ is uniquely defined for any x.

i.e. $\forall a \cdot \forall b \cdot \forall c \cdot ((a, b) \in S \ \wedge \ (a, c) \in S)) \ \rightarrow \ (b = c)$

Q2: Is the relation T a function? ✗

T *is not* a function since, for example, both $(4, 2)$ and $(4, -2)$ are elements of T (and $2 \neq -2$).

Relations vs. functions: numerical examples

Example: consider the following two relations which are both subsets of $\mathbb{Z} \times \mathbb{Z}$:

- $S \triangleq \{a \in \mathbb{Z}, b \in \mathbb{Z} \mid b = a + 1 \bullet (a, b)\}$
- $T \triangleq \{a \in \mathbb{Z}, b \in \mathbb{Z} \mid b = \sqrt{a} \bullet (a, b)\}$

Q1: Is the relation S a function? ✓

S *is* a function, since $x + 1$ is uniquely defined for any x.

i.e. $\forall\, a \cdot \forall\, b \cdot \forall\, c \cdot ((a, b) \in S \;\wedge\; (a, c) \in S)) \;\rightarrow\; (b = c)$

Q2: Is the relation T a function? ✗

T *is not* a function since, for example, both $(4, 2)$ and $(4, -2)$ are elements of T (and $2 \neq -2$).

i.e. $\exists\, a \cdot \exists\, b \cdot \exists\, c \cdot ((a, b) \in T \;\wedge\; (a, c) \in T) \;\wedge\; \neg(b = c)$

Relations vs. functions: numerical examples

Example: consider the following two relations which are both subsets of $\mathbb{Z} \times \mathbb{Z}$:

- $S \triangleq \{a \in \mathbb{Z}, b \in \mathbb{Z} \mid b = a + 1 \bullet (a, b)\}$
- $T \triangleq \{a \in \mathbb{Z}, b \in \mathbb{Z} \mid b = \sqrt{a} \bullet (a, b)\}$

Q1: Is the relation S a function? ✓

S *is* a function, since $x + 1$ is uniquely defined for any x.

i.e. $\forall\, a \cdot \forall\, b \cdot \forall\, c \cdot ((a, b) \in S \;\wedge\; (a, c) \in S)) \;\rightarrow\; (b = c)$

Q2: Is the relation T a function? ✗

T *is not* a function since, for example, both $(4, 2)$ and $(4, -2)$ are elements of T (and $2 \neq -2$).

i.e. $\exists\, a \cdot \exists\, b \cdot \exists\, c \cdot ((a, b) \in T \;\wedge\; (a, c) \in T) \;\wedge\; \neg(b = c)$

i.e. $\neg(\forall\, a \cdot \forall\, b \cdot \forall\, c \cdot ((a, b) \in T \;\wedge\; (a, c) \in T)) \;\rightarrow\; (b = c))$

Relations vs. functions: Tudor examples

If we define the set:

$$\text{Tudors} \triangleq \{\text{Henry VII, Henry VIII, Edward VI, Mary I, Elizabeth I}\}$$

Then we can define some relations (all are subsets of $\mathit{Tudors} \times \mathit{Tudors}$):

Relations vs. functions: Tudor examples

If we define the set:

Tudors $\triangleq$ {Henry VII, Henry VIII, Edward VI, Mary I, Elizabeth I }

Then we can define some relations (all are subsets of $Tudors \times Tudors$):

- is-a-parent-of = { (Henry VII, Henry VIII), (Henry VIII, Edward VI), (Henry VIII, Mary I), (Henry VIII, Elizabeth I) }

Relations vs. functions: Tudor examples

If we define the set:

Tudors $\triangleq$ {Henry VII, Henry VIII, Edward VI, Mary I, Elizabeth I }

Then we can define some relations (all are subsets of *Tudors* $\times$ *Tudors*):

- is-a-parent-of = { (Henry VII, Henry VIII), (Henry VIII, Edward VI), (Henry VIII, Mary I), (Henry VIII, Elizabeth I) }

✗ This *is not* a function since Henry VIII is a parent of three different people.

Relations vs. functions: Tudor examples

If we define the set:

$\text{Tudors} \triangleq \{$Henry VII, Henry VIII, Edward VI, Mary I, Elizabeth I $\}$

Then we can define some relations (all are subsets of $\mathit{Tudors} \times \mathit{Tudors}$):

- is-a-child-of $= \{$ (Henry VIII, Henry VII), (Edward VI, Henry VIII), (Mary I, Henry VIII), (Elizabeth I, Henry VIII) $\}$

Relations vs. functions: Tudor examples

If we define the set:

Tudors $\triangleq$ {Henry VII, Henry VIII, Edward VI, Mary I, Elizabeth I }

Then we can define some relations (all are subsets of *Tudors* $\times$ *Tudors*):

- is-a-child-of = { (Henry VIII, Henry VII), (Edward VI, Henry VIII), (Mary I, Henry VIII), (Elizabeth I, Henry VIII) }

$\checkmark$ This *is* a function since in this set everyone has a unique parent.

Relations vs. functions: Tudor examples

If we define the set:

Tudors $\triangleq$ {Henry VII, Henry VIII, Edward VI, Mary I, Elizabeth I }

Then we can define some relations (all are subsets of $Tudors \times Tudors$):

- was-succeeded-by = { (Henry VII, Henry VIII), (Henry VIII, Edward VI), (Edward VI, Mary I), (Mary I, Elizabeth I) }

Relations vs. functions: Tudor examples

If we define the set:

Tudors $\triangleq$ {Henry VII, Henry VIII, Edward VI, Mary I, Elizabeth I }

Then we can define some relations (all are subsets of *Tudors* $\times$ *Tudors*):

- was-succeeded-by = { (Henry VII, Henry VIII), (Henry VIII, Edward VI), (Edward VI, Mary I), (Mary I, Elizabeth I) }

✓ This *is* a function since each Tudor has a unique (immediate) successor.

Relations vs. functions: Tudor examples

If we define the set:

Tudors $\triangleq$ {Henry VII, Henry VIII, Edward VI, Mary I, Elizabeth I }

Then we can define some relations (all are subsets of *Tudors* $\times$ *Tudors*):

- lived-longer-than = { (Elizabeth I, Henry VII), (Elizabeth I, Henry VIII), (Elizabeth I, Mary I) (Elizabeth I, Edward VI), (Henry VIII, Henry VII), (Henry VIII, Mary I), (Henry VIII, Edward VI), (Henry VII, Mary I), (Henry VII, Edward VI), (Mary I, Edward VI) }

Relations vs. functions: Tudor examples

If we define the set:

$\text{Tudors} \triangleq \{\text{Henry VII, Henry VIII, Edward VI, Mary I, Elizabeth I}\}$

Then we can define some relations (all are subsets of $\mathit{Tudors} \times \mathit{Tudors}$):

- lived-longer-than = { (Elizabeth I, Henry VII), (Elizabeth I, Henry VIII), (Elizabeth I, Mary I) (Elizabeth I, Edward VI), (Henry VIII, Henry VII), (Henry VIII, Mary I), (Henry VIII, Edward VI), (Henry VII, Mary I), (Henry VII, Edward VI), (Mary I, Edward VI) }

This *is not* a function: for example, Elizabeth I lived longer than the other four Tudors.

Relations vs. functions: Tudor examples

If we define the set:

Tudors $\triangleq$ {Henry VII, Henry VIII, Edward VI, Mary I, Elizabeth I }

Then we can define some relations

- lived-longer-than = { (Elizabeth I, Henry VII), (Elizabeth I, Henry VIII), (Elizabeth I, Mary I) (Elizabeth I, Edward VI), (Henry VIII, Henry VII), (Henry VIII, Mary I), (Henry VIII, Edward VI), (Henry VII, Mary I), (Henry VII, Edward VI), (Mary I, Edward VI) }

This *is not* a function: for example, Elizabeth I lived longer than the other four Tudors.

Hint: since the set *Tudors* only has 5 elements, any function over this set can have *at most* five elements.

Relations vs. functions: Tudor examples

If we define the set:

$$\text{Tudors} \triangleq \{\text{Henry VII, Henry VIII, Edward VI, Mary I, Elizabeth I}\}$$

Then we can define some relations

- gender-of = { (Henry VII, male), (Henry VIII, male), (Edward VI, male), (Mary I, female) (Elizabeth I, female) }

Relations vs. functions: Tudor examples

If we define the set:

$\text{Tudors} \triangleq \{$Henry VII, Henry VIII, Edward VI, Mary I, Elizabeth I $\}$

Then we can define some relations

- gender-of $= \{$ (Henry VII, male), (Henry VIII, male), (Edward VI, male), (Mary I, female) (Elizabeth I, female) $\}$

✓ This *is* a function since each Tudor has a unique gender.

Defining functions: lambda notation

- To **define a function** we can use enumeration (for finite functions) or comprehension, since every function is a set.
- For example, we defined the function S by comprehension as:

$$S \triangleq \{a \in \mathbb{Z}, b \in \mathbb{Z} \mid b = a + 1 \bullet (a, b)\}$$

Defining functions: lambda notation

- To **define a function** we can use enumeration (for finite functions) or comprehension, since every function is a set.
- For example, we defined the function S by comprehension as:

$$S \triangleq \{a \in \mathbb{Z}, b \in \mathbb{Z} \mid b = a + 1 \bullet (a, b)\}$$

- However, since a function is a special kind of relation, we also have a special notation for defining functions.
- We can define a function much more succinctly using the lambda notation:

$$T \triangleq (\lambda\, a \in \mathbb{Z} \cdot a + 1)$$

"T is the function that takes some a, an integer, and returns $a + 1$".

Lambda notation: examples

- doubler $\triangleq (\lambda\, a \in \mathbb{N} \cdot 2 * a)$

Lambda notation: examples

- doubler $\triangleq (\lambda\, a \in \mathbb{N} \cdot 2 * a)$

  doubler is a function taking some natural number a and returning $2 * a$.

Lambda notation: examples

- doubler $\triangleq (\lambda\, a \in \mathbb{N} \cdot 2 * a)$

  doubler is a function taking some natural number a and returning $2 * a$.

  doubler $= \{(0,0),(1,2),(2,4),(3,6),\ldots\}$

Lambda notation: examples

- doubler $\triangleq (\lambda\, a \in \mathbb{N} \cdot 2 * a)$

  doubler is a function taking some natural number a and returning $2 * a$.

  doubler $= \{(0,0),(1,2),(2,4),(3,6),\ldots\}$

- pyth $\triangleq (\lambda\, a, b \in \mathbb{N} \cdot |\sqrt{a^2 + b^2}|)$

Lambda notation: examples

- doubler $\triangleq (\lambda\, a \in \mathbb{N} \cdot 2 * a)$

  doubler is a function taking some natural number a and returning $2 * a$.

  doubler $= \{(0,0), (1,2), (2,4), (3,6), \ldots\}$

- pyth $\triangleq (\lambda\, a, b \in \mathbb{N} \cdot |\sqrt{a^2 + b^2}|)$

  pyth is a function taking two natural numbers a and b, and returning the positive root $\sqrt{a^2 + b^2}$

Lambda notation: examples

- doubler $\triangleq (\lambda\, a \in \mathbb{N} \cdot 2 * a)$

  doubler is a function taking some natural number a and returning $2 * a$.

  doubler $= \{(0,0),(1,2),(2,4),(3,6),\ldots\}$

- pyth $\triangleq (\lambda\, a, b \in \mathbb{N} \cdot |\sqrt{a^2 + b^2}|)$

  pyth is a function taking two natural numbers a and b, and returning the positive root $\sqrt{a^2 + b^2}$

  pyth $= \{(3,4,5),(5,12,13),(9,40,41),\ldots\}$

Lambda notation: examples

- doubler $\triangleq (\lambda\, a \in \mathbb{N} \cdot 2 * a)$

  doubler is a function taking some natural number a and returning $2 * a$.

  doubler $= \{(0,0),(1,2),(2,4),(3,6),\ldots\}$

- pyth $\triangleq (\lambda\, a, b \in \mathbb{N} \cdot |\sqrt{a^2 + b^2}|)$

  pyth is a function taking two natural numbers a and b, and returning the positive root $\sqrt{a^2 + b^2}$

  pyth $= \{(3,4,5),(5,12,13),(9,40,41),\ldots\}$

- f $\triangleq (\lambda\, x \in \mathbb{R} \cdot x^2 + 3x + 1)$

Lambda notation: examples

- doubler $\triangleq (\lambda\, a \in \mathbb{N} \cdot 2 * a)$

  doubler is a function taking some natural number a and returning $2 * a$.

  doubler $= \{(0,0), (1,2), (2,4), (3,6), \ldots\}$

- pyth $\triangleq (\lambda\, a, b \in \mathbb{N} \cdot |\sqrt{a^2 + b^2}|)$

  pyth is a function taking two natural numbers a and b, and returning the positive root $\sqrt{a^2 + b^2}$

  pyth $= \{(3,4,5), (5,12,13), (9,40,41), \ldots\}$

- f $\triangleq (\lambda\, x \in \mathbb{R} \cdot x^2 + 3x + 1)$

  f is a function taking some real number x and returning $x^2 + 3x + 1$

Lambda notation: examples

- doubler $\triangleq (\lambda\, a \in \mathbb{N} \cdot 2 * a)$

  doubler is a function taking some natural number a and returning $2 * a$.

  doubler $= \{(0,0),(1,2),(2,4),(3,6),\ldots\}$

- pyth $\triangleq (\lambda\, a, b \in \mathbb{N} \cdot |\sqrt{a^2 + b^2}|)$

  pyth is a function taking two natural numbers a and b, and returning the positive root $\sqrt{a^2 + b^2}$

  pyth $= \{(3,4,5),(5,12,13),(9,40,41),\ldots\}$

- f $\triangleq (\lambda\, x \in \mathbb{R} \cdot x^2 + 3x + 1)$

  f is a function taking some real number x and returning $x^2 + 3x + 1$

  f $= \{(0.5, 2.75),(-0.7, 1.61),(2, 11),\ldots\}$

Lambda notation: brief history

The concept was initially developed by Gottlob Frege in a series of papers including *Function and Concept* (1891), *What is a Function?* (1904).

Lambda notation: brief history

The concept was initially developed by Gottlob Frege in a series of papers including *Function and Concept* (1891), *What is a Function?* (1904).

A system of calculating using this notation was developed by Alonzo Church into the *lambda calculus* in a series of papers including *A set of postulates for the foundation of logic*, 1932.

Lambda notation: brief history

The concept was initially developed by Gottlob Frege in a series of papers including *Function and Concept* (1891), *What is a Function?* (1904).

A system of calculating using this notation was developed by Alonzo Church into the *lambda calculus* in a series of papers including *A set of postulates for the foundation of logic*, 1932.

λ

The *lambda calculus* is the basis of functional languages such as Lisp, Scheme, FP, Miranda, ML, Haskell, Scala, F#...

Lambda notation: brief history

The concept was initially developed by Gottlob Frege in a series of papers including *Function and Concept* (1891), *What is a Function?* (1904).

A system of calculating using this notation was developed by Alonzo Church into the *lambda calculus* in a series of papers including *A set of postulates for the foundation of logic*, 1932.

λ

The *lambda calculus* is the basis of functional languages such as Lisp, Scheme, FP, Miranda, ML, Haskell, Scala, F#...

... and the basis for to anonymous functions and *lambda expressions* in Java 8 (JSR 335)

Lambda notation in Java 1.8

```
import java.util.function.*;

public class ..... {

  public static void main(String [] args) {

    IntUnaryOperator doubler = a -> 2*a;

    IntBinaryOperator pyth = (a,b) -> (int)Math.sqrt(a*a+b*b);

    DoubleUnaryOperator f = x -> x*x + 3 *x + 1;

  }
}
```