Enriched Introduction to Theory of Computation

CSC 240

Winter 2013

Faith Ellen

Logic helps us communicate precisely.

- program specifications
- database queries
- circuits

Propositions

A proposition is a statement that is either true or false.

Examples of propositions:

- $2 + 3 = 5$

Propositions

A proposition is a statement that is either true or false.

Examples of propositions:

- $2 + 3 = 5$
- $1 + 1 = 3$

Propositions

A proposition is a statement that is either true or false.

Examples of propositions:

- $2 + 3 = 5$
- $1 + 1 = 3$
- For every non-negative integer n, $n^2 + n + 41$ is prime.

Propositions

A proposition is a statement that is either true or false.

Examples of propositions:

- $2 + 3 = 5$
- $1 + 1 = 3$
- For every non-negative integer n, $n^2 + n + 41$ is prime.
- Every even integer greater than 2 is the sum of 2 prime numbers.

Ambiguous sentences:

- You may have cake or ice cream.
- If you can solve any problem in the course, then you will get an A.
- Two sisters reunited after 18 years at a checkout counter.

Connectives can change or combine propositions.

Boolean variable:

a variable that has only two possible values, true and false.

- MIT book: denoted by T and F.
- Course notes: denoted by 1 and 0.

Truth table

- columns for different combinations of Boolean variables
- one row for each possible assignment of values to the Boolean variables.

Negation

The negation of P is denoted by: NOT(P), $\neg P$, $\sim P$, $\overline{P}$

P	NOT(P)
F	T
T	F

Negation

The negation of P is denoted by: NOT(P), $\neg\mathrm{P}$, $\sim \mathrm{P}$, $\overline{\mathrm{P}}$

P	NOT(P)	NOT(NOT(P))
F	T	F
T	F	T

P and NOT(NOT(P)) are logically equivalent:
they have identical columns in the truth table.

Conjunction

The conjunction of P and Q is denoted by: P AND Q, $P \wedge Q$, $P \,\&\, Q$, P · Q

P	Q	P AND Q
F	F	F
F	T	F
T	F	F
T	T	T

Disjunction

The disjunction of P and Q is denoted by: P OR Q, $P \vee Q$, P + Q

P	Q	P OR Q
F	F	F
F	T	T
T	F	T
T	T	T

The exclusive-or of P and Q is denoted by: P XOR Q, $P \oplus Q$

P	Q	P OR Q	P XOR Q
F	F	F	F
F	T	T	T
T	F	T	T
T	T	T	F

P	Q	P OR Q	NOT (P OR Q)	NOT(P)	NOT(Q)	NOT(P) AND NOT(Q)
F	F	F	T	T	T	T
F	T	T	F	T	F	F
T	F	T	F	F	T	F
T	T	T	F	F	F	F

De Morgan's Laws:

NOT(P OR Q) and NOT(P) AND NOT(Q) are logically equivalent.

NOT(P AND Q) and NOT(P) OR NOT(Q) are logically equivalent.

```
if (x > 0 || (x <= 0 && y > 100))
```

Let A denote ‘x > 0’ and
let B denote ‘y > 100’.
A OR (NOT(A) AND B)

A	B	A OR B	NOT(A)	NOT(A) AND B	A OR (NOT(A) AND B)
F	F	F	T	F	F
F	T	T	T	T	T
T	F	T	F	F	T
T	T	T	F	F	T

A OR ((NOT A) AND B) is logically equivalent to A OR B

```
if (x > 0 || (x <= 0 && y > 100))
```

Let A denote ‘x > 0’ and
let B denote ‘y > 100’.
A OR (NOT(A) AND B)

A	B	A OR B	NOT(A)	NOT(A) AND B	A OR (NOT(A) AND B)
F	F	F	T	F	F
F	T	T	T	T	T
T	F	T	F	F	T
T	T	T	F	F	T

A OR ((NOT A) AND B) is logically equivalent to A OR B

```
if (x > 0 || y > 100)
```

Implication

Implication is denoted by: P IMPLIES Q, $P \to Q$, $P \Rightarrow Q$, $P \supset Q$

P is the hypothesis.

Q is the conclusion.

P	Q	P IMPLIES Q
F	F	T
F	T	T
T	F	F
T	T	T

P IMPLIES Q is logically equivalent to NOT(P) OR Q.

Implication

Implication is denoted by: P IMPLIES Q, $P \to Q$, $P \Rightarrow Q$, $P \supset Q$

P is the hypothesis.

Q is the conclusion.

P	Q	P IMPLIES Q	NOT(P)	NOT(P) OR Q
F	F	T	T	T
F	T	T	T	T
T	F	F	F	F
T	T	T	F	T

P IMPLIES Q is logically equivalent to NOT(P) OR Q.

Examples of P IMPLIES Q in English

P implies Q.

- Getting perfect on the final exam implies you will get an A in the course.

If P, [then] Q.

- If the list is empty, then the search is unsuccessful.
- If the input is positive, the program halts.

Q if P.

- The program halts if the input is positive.

P only if Q.

- The search is successful only if the list is nonempty.

More Examples of P IMPLIES Q in English

When[ever] P, [then] Q.

- When the robot detects an obstacle, it turns.

Q when[ever] P.

- The robot turns whenever it detects an obstacle.

P only when Q.

- The search is successful only when the list is nonempty.

Only when Q, P.

- Only when the network is connected, will the broadcast be successful.

Even More Examples of P IMPLIES Q in English

P is is sufficent/enough for Q.

- Having a good initial value is sufficient for the method to converge.

For Q, P is sufficient/enough.

- For the method to converge, a good initial value is enough .

Q is necessary for P.

- At least 40% on the final exam is necessary for passing CSC240.

For P, Q is necessary.

- To pass CSC240, at least 40% on the final exam is necessary .

P requires Q.

- A successful message receipt requires sending an acknowledgement.

NOT(Q) IMPLIES NOT(P) is the contrapositive of P IMPLIES Q

Q IMPLIES P is the converse of P IMPLIES Q

P	Q	P IMPLIES Q	NOT(P)	NOT(Q)	NOT(Q) IMPLIES NOT(P)	Q IMPLIES P
F	F	T	T	T	T	T
F	T	T	T	F	T	F
T	F	F	F	T	F	T
T	T	T	F	F	T	T

NOT(Q) IMPLIES NOT(P) is the contrapositive of P IMPLIES Q

Q IMPLIES P is the converse of P IMPLIES Q

P	Q	P IMPLIES Q	NOT(P)	NOT(Q)	NOT(Q) IMPLIES NOT(P)	Q IMPLIES P
F	F	T	T	T	T	T
F	T	T	T	F	T	F
T	F	F	F	T	F	T
T	T	T	F	F	T	T

NOT(NOT(P)) IMPLIES NOT(NOT(Q)) is the contrapositive of NOT(Q) IMPLIES NOT(P)

Equivalence

The equivalence of P and Q is denoted by:
P IFF Q, $P \leftrightarrow Q$, $P \Leftrightarrow Q$, P = Q, $P \equiv Q$
It is read as: P is equivalent to Q, P if and only if Q,
P is necessary and sufficient for Q

P	Q	P IFF Q
F	F	T
F	T	F
T	F	F
T	T	T

Predicates

A predicate is a proposition whose truth depends on the value of one or more variables.

Examples of predicates:

- $x^3 \geq 8$

Predicates

A predicate is a proposition whose truth depends on the value of one or more variables.

Examples of predicates:

- $x^3 \geq 8$
- $a^4 + b^4 + c^4 = d^4$

Predicates

A predicate is a proposition whose truth depends on the value of one or more variables.

Examples of predicates:

- $x^3 \geq 8$
- $a^4 + b^4 + c^4 = d^4$
  This predicate is true when $a = 95800$, $b = 217519$, $c = 414560$, and $d = 422481$.

Predicates

A predicate is a proposition whose truth depends on the value of one or more variables.

Examples of predicates:

- $x^3 \geq 8$
- $a^4 + b^4 + c^4 = d^4$

  This predicate is true when $a = 95800$, $b = 217519$, $c = 414560$, and $d = 422481$.
- $(x + 1)^2 > 0$

A predicate is a proposition whose truth depends on the value of one or more variables.

Examples of predicates:

- $x^3 \geq 8$
- $a^4 + b^4 + c^4 = d^4$
  This predicate is true when $a = 95800$, $b = 217519$, $c = 414560$, and $d = 422481$.
- $(x + 1)^2 > 0$
- It snowed today.

A small database:

employee	gender	salary
Andy	M	0
Donna	F	3000
Leslie	F	5000
Ron	M	5700
Tom	M	1800

Let $a(e) =$ 'employee e made at least 1000'.
Let E denote the set of employees.
Then $a : E \to \{T,F\}$.

A predicate is a function whose range is $\{T,F\}$.

Let $s(e) =$ 'salary of employee e'. This is not a predicate.
'$s(e) \geq 1000$' is a predicate.

A small database:

employee	gender	salary
Andy	M	0
Donna	F	3000
Leslie	F	5000
Ron	M	5700
Tom	M	1800

Let $a(e) =$ 'employee e made at least 1000'.

Let E denote the set of employees.

Then $a : E \to \{\text{T,F}\}$.

A predicate is a function whose range is $\{\text{T,F}\}$.

$a(e) = \text{T}$ if $e =$ Ron

$a(e) = \text{F}$ if $e =$ Andy

Every employee made at least 1000.
Each employee made at least 1000.
All employees make at least 1000.
Employees make at least 1000.

Every employee made at least 1000.
Each employee made at least 1000.
All employees make at least 1000.
Employees make at least 1000.

$\forall e \in E.a(e)$

Let $a(e)$ = 'employee e made at least 1000'.
Let E denote the set of employees.
$\forall e \in E.a(e)$

employee	gender	salary
Andy	M	0
Donna	F	3000
Leslie	F	5000
Ron	M	5700
Tom	M	1800

employee	gender	salary
Andy	M	0
Donna	F	3000
Leslie	F	5000
Ron	M	5700
Tom	M	1800

Let $a(e)$ = 'employee e made at least 1000'.
Let E denote the set of employees.
$\forall e \in E.a(e)$

This proposition is false, since Andy made less than 1000.

To show that $\forall x \in D.p(x)$ is false, give a counterexample: an $x \in D$ such that $p(x)$ is false.

Each employee made less than 6000.

Each employee made less than 6000.

Written in logic:

- Let $\ell(e) =$ 'employee e made less than 6000'.
  $\forall e \in E . \ell(e)$

Each employee made less than 6000.

Written in logic:

- Let $\ell(e)$ = 'employee e made less than 6000'.
  $\forall e \in E.\ell(e)$
- For all $e \in E$, let $\ell(e)$ = 'e made less than 6000'.
  $\forall e \in E.\ell(e)$

Each employee made less than 6000.

Written in logic:

- Let $\ell(e) =$ 'employee e made less than 6000'.
  $\forall e \in E.\ell(e)$
- For all $e \in E$, let $\ell(e) =$ 'e made less than 6000'.
  $\forall e \in E.\ell(e)$

Incorrect definition of $\ell(e)$:
Let $\ell(e) =$ 'e made less than 6000'.

Each employee made less than 6000.

Written in logic:

- Let $\ell(e) =$ 'employee e made less than 6000'.
  $\forall e \in E.\ell(e)$
- For all $e \in E$, let $\ell(e) =$ 'e made less than 6000'.
  $\forall e \in E.\ell(e)$
- $\forall e \in E.(e \text{ made less than } 6000)$

Let E denote the set of employees.

For all $e \in E$, let

$\ell(e) =$ 'e made less than 6000'.

$\forall e \in E . \ell(e)$

employee	gender	salary
Andy	M	0
Donna	F	3000
Leslie	F	5000
Ron	M	5700
Tom	M	1800

Let E denote the set of employees.

For all $e \in E$, let

$\ell(e) =$ 'e made less than 6000'.

$\forall e \in E . \ell(e)$

employee	gender	salary
Andy	M	0
Donna	F	3000
Leslie	F	5000
Ron	M	5700
Tom	M	1800

This proposition is true. Check the salary of every employee.

Let $a(e)$ = 'employee e made at least 1000'.

Let E denote the set of employees.

$\forall e \in E.\mathrm{NOT}(a(e))$

employee	gender	salary
Andy	M	0
Donna	F	3000
Leslie	F	5000
Ron	M	5700
Tom	M	1800

Let $a(e)$ = 'employee e made at least 1000'.
Let E denote the set of employees.
$\forall e \in E.\mathrm{NOT}(a(e))$

False. Tom is a counterexample.

employee	gender	salary
Andy	M	0
Donna	F	3000
Leslie	F	5000
Ron	M	5700
Tom	M	1800

Let $a(e) =$ ‘employee e made at least 1000’.
Let E denote the set of employees.
$\forall e \in E.\mathrm{NOT}(a(e))$

False. Tom is a counterexample.

$\mathrm{NOT}\,(\forall e \in E.a(e))$

employee	gender	salary
Andy	M	0
Donna	F	3000
Leslie	F	5000
Ron	M	5700
Tom	M	1800

Let $a(e)$ = 'employee e made at least 1000'
Let E denote the set of employees.
$\forall e \in E.\mathrm{NOT}(a(e))$

False. Tom is a counterexample.

$\mathrm{NOT}\,(\forall e \in E.a(e))$

True, since $\forall e \in E.a(e)$ is false.

employee	gender	salary
Andy	M	0
Donna	F	3000
Leslie	F	5000
Ron	M	5700
Tom	M	1800

Existential Quantification

There is an employee who made at least 1000.

There exists an employee who made at least 1000.

Some employee made at least 1000.

For some employee e, e made at least 1000.

At least one employee made at least 1000.

Existential Quantification

There is an employee who made at least 1000.

There exists an employee who made at least 1000.

Some employee made at least 1000.

For some employee e, e made at least 1000.

At least one employee made at least 1000.

$\exists e \in E . a(e)$

Let $a(e)$ = 'employee e made at least 1000'.
Let E denote the set of employees
$\exists e \in E . a(e)$

employee	gender	salary
Andy	M	0
Donna	F	3000
Leslie	F	5000
Ron	M	5700
Tom	M	1800

employee	gender	salary
Andy	M	0
Donna	F	3000
Leslie	F	5000
Ron	M	5700
Tom	M	1800

Let $a(e)$ = 'employee e made at least 1000'.
Let E denote the set of employees.
$\exists e \in E . a(e)$

This proposition is true. Tom is an example that demonstrates this.

To show that $\exists x \in D . p(x)$ is true, give an example:
an $x \in D$ such that $p(x)$ is true.

employee	gender	salary
Andy	M	0
Donna	F	3000
Leslie	F	5000
Ron	M	5700
Tom	M	1800

Let $a(e)$ = 'employee e made at least 1000'.
Let E denote the set of employees.
$\exists e \in E.a(e)$

This proposition is true. Tom is an example that demonstrates this.

To show that $\exists x \in D.p(x)$ is true, give an example:
an $x \in D$ such that $p(x)$ is true.

Provided $D \neq \phi$,
if $\forall x \in D.p(x)$ is true, then $\exists x \in D.p(x)$ is true.

Some employee made at least 6000.

Some employee made at least 6000.

Written in logic:

- Let $g(e)$ = 'employee e made at least 6000'.
  $\exists e \in E . g(e)$

Some employee made at least 6000.

Written in logic:

- Let $g(e)$ = 'employee e made at least 6000'.
  $\exists e \in E . g(e)$
- $\exists e \in E . \mathrm{NOT}(\ell(e))$

Let E denote the set of employees.

For all $e \in E$, let

$g(e) =$ ‘e made at least 6000’.

$\exists e \in E . g(e)$

employee	gender	salary
Andy	M	0
Donna	F	3000
Leslie	F	5000
Ron	M	5700
Tom	M	1800

Let E denote the set of employees.

For all $e \in E$, let

$g(e) =$ 'e made at least 6000'.

$\exists e \in E . g(e)$

employee	gender	salary
Andy	M	0
Donna	F	3000
Leslie	F	5000
Ron	M	5700
Tom	M	1800

This proposition is false. Check the salary of every employee.

$\exists x \in D.p(x)$ is false when $p(x)$ is false for every $x \in D$.

$\mathrm{NOT}\,(\exists x \in D.p(x))$ is logically equivalent to $\forall x \in D.\mathrm{NOT}(p(x))$.

$\mathrm{NOT}\,(\exists x \in D.p(x))$ IFF $\forall x \in D.\mathrm{NOT}(p(x))$ is true.

Quantification with Negation

$\exists x \in D.p(x)$ is false when $p(x)$ is false for every $x \in D$.

$\mathrm{NOT}\,(\exists x \in D.p(x))$ is logically equivalent to $\forall x \in D.\mathrm{NOT}(p(x))$.

$\mathrm{NOT}\,(\exists x \in D.p(x))$ IFF $\forall x \in D.\mathrm{NOT}(p(x))$ is true.

$\forall x \in D.p(x)$ is false when there exists a counterexample.

$\mathrm{NOT}\,(\forall x \in D.p(x))$ IFF $\exists x \in D.\mathrm{NOT}(p(x))$ is true.

An employee made at least 1000.

An employee made at least 1000.

Written in logic:

- $\forall e \in E.a(e)$
- $\exists e \in E.a(e)$

Universal Quantification with Disjunction

Every employee made at least 1000 or every employee made less than 5000.

employee	gender	salary
Andy	M	0
Donna	F	3000
Leslie	F	5000
Ron	M	5700
Tom	M	1800

Universal Quantification with Disjunction

Every employee made at least 1000 or
every employee made less than 5000.

Let $b(e) =$
 'employee e made less than 5000'.

$(\forall e \in E.a(e))$ OR $(\forall e \in E.b(e))$

false
Andy is a counterexample to $\forall e \in E.a(e)$
Ron is a counterexample to $\forall e \in E.b(e)$

employee	gender	salary
Andy	M	0
Donna	F	3000
Leslie	F	5000
Ron	M	5700
Tom	M	1800

Universal Quantification with Disjunction

Every employee made at least 1000 or every employee made less than 5000.

$(\forall e \in E.a(e))$ OR $(\forall e \in E.b(e))$

Every employee made at least 1000 or less than 5000.

$\forall e \in E.(a(e)$ OR $b(e))$

employee	gender	salary
Andy	M	0
Donna	F	3000
Leslie	F	5000
Ron	M	5700
Tom	M	1800

Universal Quantification with Disjunction

Every employee made at least 1000 or every employee made less than 5000.

$(\forall e \in E.a(e))$ OR $(\forall e \in E.b(e))$

Every employee made at least 1000 or less than 5000.

$\forall e \in E.(a(e)$ OR $b(e))$

employee	gender	salary
Andy	M	0
Donna	F	3000
Leslie	F	5000
Ron	M	5700
Tom	M	1800

If $(\forall x \in D.p(x))$ OR $(\forall x \in D.q(x))$ is true, then $\forall x \in D.(p(x)$ OR $q(x))$ is true.

The converse is not always true.

Every employee made at least 1000 and every employee made less than 5000.

Every employee made at least 1000 and less than 5000.

employee	gender	salary
Andy	M	0
Donna	F	3000
Leslie	F	5000
Ron	M	5700
Tom	M	1800

Universal Quantification with Conjunction

Every employee made at least 1000 and every employee made less than 5000.

Every employee made at least 1000 and less than 5000.

employee	gender	salary
Andy	M	0
Donna	F	3000
Leslie	F	5000
Ron	M	5700
Tom	M	1800

$(\forall x \in D.p(x))$ AND $(\forall x \in D.q(x))$ is equivalent to $\forall x \in D.(p(x)$ AND $q(x))$.

Some employee made at least 1000 and less than 1500.

Some employee made at least 1000 and some employee made less than 1500.

employee	gender	salary
Andy	M	0
Donna	F	3000
Leslie	F	5000
Ron	M	5700
Tom	M	1800

Existential Quantification with Conjunction

Some employee made at least 1000 and less than 1500.

Some employee made at least 1000 and some employee made less than 1500.

employee	gender	salary
Andy	M	0
Donna	F	3000
Leslie	F	5000
Ron	M	5700
Tom	M	1800

If $\exists x \in D.(p(x) \text{ AND } q(x))$ is true,
then $(\exists x \in D.p(x))$ AND $(\exists x \in D.q(x))$ is true.
The converse is not always true.

Existential Quantification with Conjunction

Some employee made at least 1000 and less than 1500.

Some employee made at least 1000 and some employee made less than 1500.

employee	gender	salary
Andy	M	0
Donna	F	3000
Leslie	F	5000
Ron	M	5700
Tom	M	1800

If $\exists x \in D.(p(x) \text{ AND } q(x))$ is true,
then $(\exists x \in D.p(x)) \text{ AND } (\exists x \in D.q(x))$ is true.
The converse is not always true.

$(\exists x \in D.p(x)) \text{ AND } (\exists y \in D.q(y))$

Existential Quantification with Conjunction

Some employee made at least 1000 and less than 1500.

Some employee made at least 1000 and some employee made less than 1500.

employee	gender	salary
Andy	M	0
Donna	F	3000
Leslie	F	5000
Ron	M	5700
Tom	M	1800

If $\exists x \in D.(p(x) \text{ AND } q(x))$ is true,
then $(\exists x \in D.p(x))$ AND $(\exists x \in D.q(x))$ is true.
The converse is not always true.

$(\exists x \in D.p(x))$ AND $(\exists y \in D.q(y))$ is equivalent to
$\exists x \in D.\exists y \in D.(p(x) \text{ AND } q(y))$

$(\forall x \in D.p(x))$ OR $(\forall y \in D.q(y))$ is equivalent to
$\forall x \in D.\forall y \in D.(p(x) \text{ OR } q(y))$

Existential Quantification with Disjunction

$\exists x \in D.(p(x) \text{ OR } q(x))$ is equivalent to
$(\exists x \in D.p(x)) \text{ OR } (\exists x \in D.q(x))$.

More Quantification

Some female employee made at least 1000.

E = the set of employees.
$a(e)$ = 'employee e made at least 1000'.

More Quantification

Some female employee made at least 1000.

E = the set of employees.
$a(e)$ = 'employee e made at least 1000'.

- Let F = the set of female employees.
  $\exists e \in F.a(e)$
- For $e \in E$, let $f(e)$ = 'e is female'.
  $\exists e \in E.(f(e) \text{ AND } a(e))$
- Let A = the set of employees who made at least 1000.
  Let $f(e)$ = 'employee e is female'.
  $\exists e \in A.f(e)$

All female employees made at least 1000.

$E =$ the set of employees.

$A =$ the set of employees who made at least 1000.

$F =$ the set of female employees.

$a(e) =$ 'employee e made at least 1000'.

$f(e) =$ 'employee e is female'.

All female employees made at least 1000.

E = the set of employees.
A = the set of employees who made at least 1000.
F = the set of female employees.

$a(e)$ = 'employee e made at least 1000'.
$f(e)$ = 'employee e is female'.

- $\forall e \in F . a(e)$

All female employees made at least 1000.

For all employees, if the employee is female,
then she made at least 1000.

If an employee is female, then she made at least 1000.

E = the set of employees.
A = the set of employees who made at least 1000.
F = the set of female employees.
$a(e)$ = 'employee e made at least 1000'.
$f(e)$ = 'employee e is female'.

- $\forall e \in F.a(e)$
- $\forall e \in E.(f(e)$ IMPLIES $a(e))$

All female employees made at least 1000.

For all employees, if the employee is female,
  then she made at least 1000.

If an employee is female, then she made at least 1000.

E = the set of employees.
A = the set of employees who made at least 1000.
F = the set of female employees.
$a(e)$ = 'employee e made at least 1000'.
$f(e)$ = 'employee e is female'.

- $\forall e \in F. a(e)$
- $\forall e \in E. (f(e) \text{ IMPLIES } a(e))$
- $\forall e \in E. (e \in F \text{ IMPLIES } e \in A)$.

Some x with property p also has property q.
$\exists x \in D.(p(x) \text{ AND } q(x))$

Every x with property p also has property q.
$\forall x \in D.(p(x) \text{ IMPLIES } q(x))$

Some x with property p also has property q.
$\exists x \in D.(p(x) \text{ AND } q(x))$

Every x with property p also has property q.
$\forall x \in D.(p(x) \text{ IMPLIES } q(x))$

There is no x with property p that also has property q.
Every x with property p does not have property q.
Not every x with property p also has property q.
There is some x with property p that does not have property q.

Some x with property p also has property q.
$\exists x \in D.(p(x) \text{ AND } q(x))$

Every x with property p also has property q.
$\forall x \in D.(p(x) \text{ IMPLIES } q(x))$

There is no x with property p that also has property q.
$\text{NOT}(\exists x \in D.(p(x) \text{ AND } q(x)))$

Every x with property p does not have property q.
$\forall x \in D.(p(x) \text{ IMPLIES NOT}(q(x)))$

Not every x with property p also has property q.
$\text{NOT}(\forall x \in D.(p(x) \text{ IMPLIES } q(x)))$

There is some x with property p that does not have property q.
$\exists x \in D.(p(x) \text{ AND NOT}(q(x)))$

All employees that made at least 6000 are female.

employee	gender	salary
Andy	M	0
Donna	F	3000
Leslie	F	5000
Ron	M	5700
Tom	M	1800

Let $E =$ the set of employees.
Let $g(e) =$ 'employee e made at least 6000'.
Let $f(e) =$ 'employee e is female'.

$\forall e \in E.(g(e) \text{ IMPLIES } f(e))$

All employees that made at least 6000 are female.

Let E = the set of employees.
Let $g(e)$ = 'employee e made at least 6000'.
Let $f(e)$ = 'employee e is female'.

$\forall e \in E.(g(e)$ IMPLIES $f(e))$

$\forall e \in E.($NOT$(f(e))$ IMPLIES NOT$(g(e)))$

All male employees made less than 6000.

employee	gender	salary
Andy	M	0
Donna	F	3000
Leslie	F	5000
Ron	M	5700
Tom	M	1800

All employees that made at least 6000 are female.

Let E = the set of employees.

Let $g(e)$ = 'employee e made at least 6000'.

Let $f(e)$ = 'employee e is female'.

$\forall e \in E.(g(e) \text{ IMPLIES } f(e))$

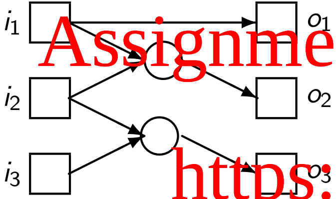
$\forall e \in E.(\text{NOT}(f(e)) \text{ IMPLIES } \text{NOT}(g(e)))$

employee	gender	salary
Andy	M	0
Donna	F	3000
Leslie	F	5000
Ron	M	5700
Tom	M	1800

All male employees made less than 6000.

$\forall e \in E.(g(e) \text{ IMPLIES } f(e))$ is vacuously true.

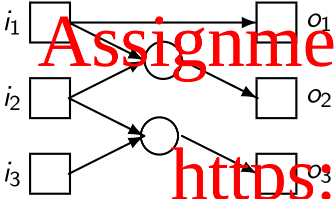
Mixing Quantifiers



Let $I = \{i_1, i_2, i_3\}$.
Let $O = \{o_1, o_2, o_3\}$.
Let $c(i, o) =$ 'input i connects to output o'.

$\forall o \in O. \exists i \in I. c(i, o)$
Every output has some input that connects to it. true

$\exists i \in I. \forall o \in O. c(i, o)$
Some input connects to all outputs. false

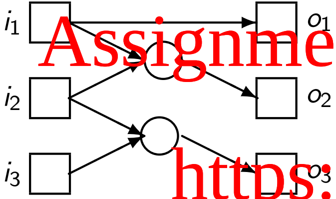


Let $I = \{i_1, i_2, i_3\}$.
Let $O = \{o_1, o_2, o_3\}$.
Let $c(i, o) =$ 'input i connects to output o'.

$\forall o \in O.\exists i \in I.c(i, o)$
Every output has some input that connects to it. true

$\exists i \in I.\forall o \in O.c(i, o)$
Some input connects to all outputs. false

$(\exists x \in D.\forall y \in D'.p(x, y))$ IMPLIES $(\forall y \in D'.\exists x \in D.p(x, y))$



Let $I = \{i_1, i_2, i_3\}$.
Let $O = \{o_1, o_2, o_3\}$.
Let $c(i, o) =$ 'input i connects to output o'.

$\forall o \in O.\exists i \in I.c(i, o)$
Every output has some input that connects to it. true

$\exists i \in I.\forall o \in O.c(i, o)$
Some input connects to all outputs. false

$(\exists x \in D.\forall y \in D'.p(x, y))$ IMPLIES $(\forall y \in D'.\exists x \in D.p(x, y))$

$\forall y \in D'.\exists x_y \in D.p(x_y, y)$

$\lim_{x \to a} f(x) = d$

$\forall \epsilon \in \mathbb{R}^+ . \exists \delta \in \mathbb{R}^+ . \forall x \in \mathbb{R}. \, (|x - a| < \delta \text{ IMPLIES } |f(x) - d| < \epsilon)$

$f \in O(g)$

$\exists c \in \mathbb{R}^+ . \exists b \in \mathbb{N}. \, (x \geq b \text{ IMPLIES } f(x) \leq cg(x))$

$f \in \Omega(g)$

$\exists c \in \mathbb{R}^+ . \exists b \in \mathbb{N}. \, (x \geq b \text{ IMPLIES } f(x) \geq cg(x))$

Let $M : R \times C \to \{\text{T,F}\}$

Every row of M contains T.

Let $M : R \times C \to \{\mathrm{T},\mathrm{F}\}$

Every row of M contains T.

$\forall x \in R. \exists y \in C. M(x, y)$

Let $M : R \times C \to \{\mathrm{T,F}\}$

Every row of M contains T.
$\forall x \in R. \exists y \in C. M(x, y)$

$\mathrm{NOT}\,(\forall x \in R. \exists y \in C. M(x, y))$
Not every row of M contains T.

$\exists x \in R. \mathrm{NOT}\,(\exists y \in C. M(x, y))$
Some row of M does not contain T.

$\exists x \in R. \forall y \in C. \mathrm{NOT}(M(x, y))$
Some row of M is all F.

Let $M : R \times C \to \{\text{T,F}\}$

Every row of M contains T.
$\forall x \in R.\exists y \in C.M(x, y)$

NOT $(\forall x \in R.\exists y \in C.M(x, y))$
Not every row of M contains T.

$\exists x \in R.\text{NOT}\,(\exists y \in C.M(x, y))$
Some row of M does not contain T.

$\exists x \in R.\forall y \in C.\text{NOT}(M(x, y))$
Some row of M is all F.

NOT $(\exists y \in C.\forall x \in R.M(x, y))$ is equivalent to $\forall y \in C.\exists x \in R.\text{NOT}(M(x, y))$

The negation of 'Some column of M is entirely T' is
'Every column of M contains F'.

Problem 1

Write the following sentence in logic, using the notation from the course slides (and MIT book):

The lockdown won't end unless the infection rate has decreased.

Problem 1

Write the following sentence in logic, using the notation from the course slides (and MIT book):

The lockdown won't end unless the infection rate has decreased.

Let L denote "The lockdown will end".
Let D denote "The infection rate has decreased".
(NOT L) UNLESS D

The lockdown will not end unless the infection rate has decreased.

Let L denote "The lockdown will end".
Let D denote "The infection rate has decreased".
Which of the following mean the same as (NOT L) UNLESS D?

1. If the infection rate has decreased, the lockdown will end.
   D IMPLIES L
2. If the infection rate has decreased, the lockdown will not end.
   D IMPLIES (NOT L) or, equivalently, L IMPLIES (NOT D)
3. If the infection rate has not decreased, the lockdown will end.
   (NOT D) IMPLIES L or, equivalently, (NOT L) IMPLIES D
4. If the infection rate has not decreased, the lockdown will not end.
   (NOT D) IMPLIES (NOT L) or, equivalently, L IMPLIES D

The lockdown will not end unless the infection rate has decreased.

Let L denote "The lockdown will end".
Let D denote "The infection rate has decreased".

If the infection rate has not decreased, the lockdown will not end.
(NOT L) UNLESS D means the same as L IMPLIES D
If the lockdown ends, the infection rate has decreased.

(NOT P) UNLESS Q means the same as P IMPLIES Q

I won't give you cake unless you have eaten your vegetables.

(NOT P) UNLESS Q means the same as P IMPLIES Q

I won't give you cake unless you have eaten your vegetables.

Let P denote "I will give you cake".

Let Q denote "you have eaten your vegetables".

(NOT P) UNLESS Q means the same as P IMPLIES Q

I won't give you cake unless you have eaten your vegetables.

Let P denote "I will give you cake".
Let Q denote "you have eaten your vegetables".

Suppose the child eats their vegetables.
If the parent gives doesn't give the child cake, did the parent lie?

(NOT P) UNLESS Q means the same as P IMPLIES Q

I won't give you cake unless you have eaten your vegetables.

Let P denote "I will give you cake".
Let Q denote "you have eaten your vegetables".

Suppose the child eats their vegetables.
If the parent gives doesn't give the child cake, did the parent lie?

P IFF Q

(NOT P) UNLESS Q means the same as P IMPLIES Q

I won't give you cake unless you have eaten your vegetables.

Let P denote "I will give you cake".
Let Q denote "you have eaten your vegetables".

Suppose the child eats their vegetables.
If the parent gives doesn't give the child cake, did the parent lie?

P IFF Q

avoid using UNLESS, DESPITE, NOT WITHSTANDING

Problem 2

Consider a circuit with three Boolean inputs, x_1, x_0, and b and three Boolean outputs, c, y_1 and y_0, where the string cy_1y_0 represents the sum of b and the number represented by the string x_1x_0.

$$\begin{array}{cccc} & & x_1 & x_0 \\ + & & & b \\ \hline & c & y_1 & y_0 \end{array}$$

(a) Express each output as a combination of the inputs using logical connectives.

Problem 2

Consider a circuit with three Boolean inputs, x_1, x_0, and b and three Boolean outputs, c, y_1 and y_0, where the string cy_1y_0 represents the sum of b and the number represented by the string x_1x_0.

$$\begin{array}{ccc} & x_1 & x_0 \\ + & & b \\ \hline c & y_1 & y_0 \end{array}$$

(a) Express each output as a combination of the inputs using logical connectives.

Generalize your circuit to have $n+1$ inputs, $x_{n-1}, \ldots, x_0$, b and $n+1$ outputs, c, $y_{n-1}, \ldots, y_0$.

Consider a circuit with three Boolean inputs, x_1, x_0, and b and three Boolean outputs, c, y_1, and y_0, where the string cy_1y_0 represents the sum of b and the number represented by the string x_1x_0.

(a) Express each output as a combination of the inputs, using logical connectives.

$y_0 = x_0$ XOR b

$y_1 = x_1$ XOR (x_0 AND b)

$c = x_1$ AND x_0 AND b

This is called a 2-bit half-adder.

Consider a circuit with three Boolean inputs, x_1, x_0, and b and three Boolean outputs, c, y_1, and y_0, where the string cy_1y_0 represents the sum of b and the number represented by the string x_1x_0.

(a) Express each output as a combination of the inputs, using logical connectives.

$y_0 = x_0$ XOR b

$y_1 = x_1$ XOR (x_0 AND b)

$c = x_1$ AND x_0 AND b

AND is associative:

(A AND B) AND C is logically equivalent to A AND (B AND C) so brackets are not needed, i.e. it is fine to write A AND B AND C.

AND is associative:

(A AND B) AND C is logically equivalent to A AND (B AND C).

Which among OR, XOR, IMPLIES, and IFF are associative?

AND is associative:
(A AND B) AND C is logically equivalent to A AND (B AND C).

Which among OR, XOR, IMPLIES, and IFF are associative?

F IMPLIES (F IMPLIES F) = T, but
(F IMPLIES F) IMPLIES F = F.
Therefore IMPLIES is not associative.

OR, XOR, and IFF are all associative.

(b) Generalize your circuit to have $n+1$ inputs, $x_{n-1}, \ldots, x_0$, b and $n+1$ outputs, c, $y_{n-1}, \ldots, y_0$.

$y_0 = x_0$ XOR b

$c_1 = x_0$ AND b

$y_i = x_i$ XOR c_i, for $i = 1, \ldots, n-1$

$c_{i+1} = x_i$ AND c_i, for $i = 1, \ldots, n-1$

$c = c_n$

The idea is that c_i denotes the carry into column i from column $i-1$, where the columns are numbered from right to left, starting with 0.

This can be written more succinctly, as follows:

$c_0 = b$

$y_i = x_i$ XOR c_i, for $i = 0, \ldots, n-1$

$c_{i+1} = x_i$ AND c_i, for $i = 0, \ldots, n-1$

$c = c_n$

Problem 5

Which of the following statements are true and which are false? Explain.

a. $\forall x \in \mathbb{N}. \exists y \in \mathbb{N}. (2x - y = 0)$

b. $\exists y \in \mathbb{N}. \forall x \in \mathbb{N}. (2x - y = 0)$

c. $\forall x \in \mathbb{N}. (x < 10 \text{ IMPLIES } \forall y \in \mathbb{N}. (y < x \text{ IMPLIES } y < 9))$

a. $\forall x \in \mathbb{N}. \exists y \in \mathbb{N}. (2x - y = 0)$

a. $\forall x \in \mathbb{N}. \exists y \in \mathbb{N}. (2x - y = 0)$ True

Given $x \in \mathbb{N}$, let $y = 2x$.

a. $\forall x \in \mathbb{N}. \exists y \in \mathbb{N}. (2x - y = 0)$ True

Given $x \in \mathbb{N}$, let $y = 2x$.

b. $\exists y \in \mathbb{N}. \forall x \in \mathbb{N}. (2x - y = 0)$

a. $\forall x \in \mathbb{N}. \exists y \in \mathbb{N}. (2x - y = 0)$ True
Given $x \in \mathbb{N}$, let $y = 2x$.

b. $\exists y \in \mathbb{N}. \forall x \in \mathbb{N}. (2x - y = 0)$ False
Given $y \in \mathbb{N}$, let $x = y + 1$. Then
$2x - y = 2(y + 1) - y = y + 2 > 0$.

a. $\forall x \in \mathbb{N}.\exists y \in \mathbb{N}.(2x - y = 0)$ True

Given $x \in \mathbb{N}$, let $y = 2x$.

b. $\exists y \in \mathbb{N}.\forall x \in \mathbb{N}.(2x - y = 0)$ False

Given $y \in \mathbb{N}$, let $x = y + 1$. Then $2x - y = 2(y + 1) - y = y + 2 > 0$.

c. $\forall x \in \mathbb{N}.(x < 10 \text{ IMPLIES } \forall y \in \mathbb{N}.(y < x \text{ IMPLIES } y < 9))$

a. $\forall x \in \mathbb{N}.\exists y \in \mathbb{N}.(2x - y = 0)$ True

Given $x \in \mathbb{N}$, let $y = 2x$.

b. $\exists y \in \mathbb{N}.\forall x \in \mathbb{N}.(2x - y = 0)$ False

Given $y \in \mathbb{N}$, let $x = y + 1$. Then

$2x - y = 2(y + 1) - y = y + 2 > 0$.

c. $\forall x \in \mathbb{N}.(x < 10 \text{ IMPLIES } \forall y \in \mathbb{N}.(y < x \text{ IMPLIES } y < 9))$

True

If $x < 10$ then $x \leq 9$. Hence, if $y < x$, then $y < 9$.

Problem 6

Write a predicate *floor* : $\mathbb{R} \times \mathbb{Z} \to \{T,F\}$ such that
for $x \in \mathbb{R}$ and $y \in \mathbb{Z}$,
$floor(x, y) = T$ if y is the largest integer less than or equal to x.

Problem 6

Write a predicate *floor* : $\mathbb{R} \times \mathbb{Z} \to \{\text{T,F}\}$ such that
for $x \in \mathbb{R}$ and $y \in \mathbb{Z}$,
$floor(x, y) = \text{T}$ if y is the largest integer less than or equal to x.

In definitions, the English word 'if' often means 'if and only if'.

Problem 6

Write a predicate *floor* : $\mathbb{R} \times \mathbb{Z} \rightarrow \{\text{T,F}\}$ such that

for $x \in \mathbb{R}$ and $y \in \mathbb{Z}$,

$$floor(x, y) = \begin{cases} \text{T} & \text{if y is the largest integer less than or equal to x.} \\ \text{F} & \text{otherwise.} \end{cases}$$

Problem 6

Write a predicate *floor* : $\mathbb{R} \times \mathbb{Z} \to \{T,F\}$ such that
for $x \in \mathbb{R}$ and $y \in \mathbb{Z}$,
$floor(x, y) =$ T if y is the largest integer less than or equal to x.

Write a predicate *round* : $\mathbb{R} \times \mathbb{Z} \to \{T,F\}$ such that
for $x \in \mathbb{R}$ and $y \in \mathbb{Z}$,
$round(x, y) =$ T if y is the closest integer to x.

Problem 6

Write a predicate *floor* : $\mathbb{R} \times \mathbb{Z} \to \{T,F\}$ such that
for $x \in \mathbb{R}$ and $y \in \mathbb{Z}$,
floor$(x, y) = T$ if y is the largest integer less than or equal to x.

For $x \in \mathbb{R}$ and $y \in \mathbb{Z}$, let
floor$(x, y) =$ "$(y \leq x)$ AND $\forall z \in \mathbb{Z}.[(z \leq x)$ IMPLIES $(z \leq y)]$"

Problem 6

Write a predicate *floor* : $\mathbb{R} \times \mathbb{Z} \rightarrow \{T,F\}$ such that
for $x \in \mathbb{R}$ and $y \in \mathbb{Z}$,
floor$(x, y) =$ T if y is the largest integer less than or equal to x.

For $x \in \mathbb{R}$ and $y \in \mathbb{Z}$, let
$floor(x, y) =$ "$(y \leq x)$ AND $\forall z \in \mathbb{Z}.[(z \leq x)$ IMPLIES $(z \leq y)]$"

For $x \in \mathbb{R}$ and $y \in \mathbb{Z}$, let
$floor(x, y) =$ "$(y \leq x)$ AND $(x < y + 1)$"

Write a predicate *round* : $\mathbb{R} \times \mathbb{Z} \to \{\text{T,F}\}$ such that for $x \in \mathbb{R}$ and $y \in \mathbb{Z}$

$round(x, y) = \text{T}$ if y is the closest integer to x.

For $x \in \mathbb{R}$ and $y \in \mathbb{Z}$, let
$round(x, y) =$ "$\forall z \in \mathbb{Z}.(|y - x| \leq |z - x|)$" or, alternatively,
$round(x, y) =$ "$|y - x| \leq .5$"

Note that both $round(.5, 1) = \text{T}$ and $round(.5, 0) = \text{T}$.
Modify this predicate so that if $x = y + .5$ for some integer y, then $round(x, y + 1) = \text{T}$, but $round(x, y) = \text{F}$.

TA: Kayman :)

- Common Mistakes
- Quiz
- Problem Session
- Questions

Common Mistakes

BAD: f = faulty, no x!

Let $f(x)$ = "faulty".

Let $f(x)$ = "process q is faulty"

(1) Unquantified variables appear on both sides of the equation.

BAD! $f(x)$ = "person x is female"

$\forall$ mark $\in$ People . Not(f(mark)) $\quad f(x) = \forall x \in P$

(2) Don't quantify constants or unquantified variables.

$\exists$ Not(f(mark)).

BAD: $f(x)$ = "x is faulty"

GOOD: Let P = "set of processes"

$\forall x \in P$, let $f(x)$ = "x is faulty"

$f : P \to \{T, F\}$.

BAD: Let x = "all process"

$f(y)$ = "y is faulty"

$NOT(f(x))$ —

CORRECT Let P = "all process"

$\forall x \in P.\ NOT(f(x))$.

$\forall x \in P.\ q(x),\ p(x)$

④ Predicates must be combined with quantifiers.

$q(x)$ AND $p(x)$

BAD! $\forall x, y \in P.\ f(x)$ AND $f(y)$ ←

$\forall x \in P.\ \forall y \in P.\ f(x)$ AND $f(y)$

⑤ Each quantified variable needs its own quantifier, and each quantifier needs its own domain.

$\forall x, y \in A$? **NO**

$f(x)$ = "x is false".

$\forall x \in P.\ f(x)$ is false ← BAD

$\forall x \in P.\ NOT(f(x))$

Question from CSC240 Regular Tutorial 1
TA: Kayman Brusse

Problem

Translate the following English sentences into logic, using only the two predicates:
$t(x, y, z) =$ "process x transmitted process y's message to process z", and
$w(x, y) =$ "process x wants to send a message to process y".
q denotes a particular process.
Use brackets where necessary to avoid any possible ambiguity.

a. No process transmitted q's message, but some process wants to send a message to q
b. Every process transmitted its message to a process that does not want to send it a message.
c. Some process wants to send a message to every process that transmitted its message.

$t(x, y, z) =$ "process x transmitted process y's message to process z", and
$w(x, y) =$ "process x wants to send a message to process y".

c. Some process wants to send a message to every process that transmitted its message.

Let P denote the set of processes.
One translation is
$\exists y \in P.\forall x \in P.\forall z \in P.(t(x, y, z) \text{ IMPLIES } w(y, x))$,
or, equivalently,
$\exists y \in P.\forall x \in P.((\exists z \in P.t(x, y, z)) \text{ IMPLIES } w(y, x))$.

Question: Why are the solutions to part (c) equivalent? Why can you bring the introduction of z into the implication, and why does the quantifier change?

Answer: In tutorial, I said: *because the variable z does not appear in the consequent $w(y, x)$ of the implication.*

This was not a very helpful response, so here is a more complete answer. It suffices to see that the statements

$$\forall z \in P.(t(x, y, z) \text{ IMPLIES } w(y, x)) \quad \text{and} \quad ((\exists z \in P.t(x, y, z)) \text{ IMPLIES } w(y, z))$$

are equivalent. There are laws which allow you to transform one statement into another, such that both statements are logically equivalent. These laws are covered in Video Lecture 4. In particular (see timestamp 10:30), the law that can be used here is the following: If D is a set, $p(x)$ is a predicate and E is a formula such that the variable x does not occur in E then

$$\forall x \in D.(p(x) \text{ IMPLIES } E) \text{ is logically equivalent to } (\exists x \in D.p(x)) \text{ IMPLIES } E$$

As an exercise, once you have watched Lecture 4 and understand what it means for two formulas to be logically equivalent, you should try to *derive* that this law from other transformations. That is, give a sequence of transformations that turn the formula on the left into the formula on the right. Doing this will help you understand *why* the quantifier needs to change when you bring the formula inside the implication.

This law is also discussed in the Course Notes for CSCB38/236/240 (see **IIe** on page 160). These notes use different notation, but they show how to derive this law from other transformations.

Solutions to

CSC240 Winter 2021 Homework Assignment 1

1. (a) Every integer in S occurs infinitely many times.
   (b) $S = (S_1, S_2, \ldots)$ where $S_i = 2$ for all $i \geq 1$.
   (c) $S = (S_1, S_2, \ldots)$ where $S_i = i$ for all $i \geq 1$.

2. Let $\mathcal{P}$ denote the set of all functions from $D \times D$ into $D \cup \{\Box\}$. Note that $\mathcal{P}$ can also be written as $(D \cup \{\Box\})^{D \times D}$.

   For any two Sukodu puzzles P and A, let $consistent : \mathcal{P} \times \mathcal{P} \to \{\text{T,F}\}$ denote the predicate such that $consistent(P, A)$ is true when every nonempty grid location in P has the same value in A.
   So, $consistent(P, A) = \forall i \in D. \forall j \in D. (\text{NOT}(P[i,j] = \Box) \text{ IMPLIES } (P[i,j] = A[i,j]))$.

   For any $A \in \mathcal{P}$, let $rowsOK : \mathcal{P} \to \{\text{T,F}\}$ denote the predicate such that $rowsOK(A)$ is T if and only if every row of A contains each of the numbers in D.
   Then $rowsOK(A) = \forall i \in D. \forall d \in D. \exists j \in D. (A[i,j] = d)$.

   Similarly, let $columnsOK : \mathcal{P} \to \{\text{T,F}\}$ denote the predicate such that $columnsOK(A)$ is T if and only if every column of A contains each of the numbers in D.
   Then $columnsOK(A) = \forall j \in D. \forall d \in D. \exists i \in D. (A[i,j] = d)$.

   Finally, we let $subgridsOK : \mathcal{P} \to \{\text{T,F}\}$ denote the predicate such that $subgridsOK(A)$ is T if and only if each of the k^2 disjoint subgrids of A contains each of the numbers in D.
   To write this predicate, let $M = \{i \in \mathbb{Z} \mid 1 \leq i \leq k\}$ denote the rows and columns in the upper leftmost subgrid and let $L = \{u \cdot k \mid (u \in \mathbb{N}) \text{ AND } (0 \leq u \leq k-1)\}$ denote the possible row and column offsets for each of the other subgrids. Then, for each $r, c \in L$, $\{(r+i, c+j) \mid i, j \in M\}$ is the set of the grid locations of the $k \times k$ subgrid whose topmost row is $r+1$ and whose leftmost column is $c+1$.
   Hence $subgridsOK(A) = \forall r \in L. \forall c \in L. \forall d \in D. \exists i \in M. \exists j \in M. A[r+i, c+j] = d$.

   Combining all these, we get the predicate $correct : \mathcal{P} \times \mathcal{P} \to \{\text{T,F}\}$, where $correct(P, A) = consistent(P, A) \text{ AND } rowsOK(A) \text{ AND } columnsOK(A) \text{ AND } subgridsOK(A)$.

Validity

A propositional formula is an expression built up from Boolean variables using connectives such as
AND, OR, NOT, IMPLIES, IFF, XOR.
It does not contain predicates or quantifiers.

A propositional formula is valid or a tautology
if all its truth table entries are true.

Examples:

- P OR NOT(P)
- NOT(P OR Q) IFF (NOT(P) AND NOT(Q))

Validity

A propositional formula is an expression built up from Boolean variables using connectives such as
AND, OR, NOT, IMPLIES, IFF, XOR.
It does not contain predicates or quantifiers.

A propositional formula is valid or a tautology
if all its truth table entries are true.

Examples:

- P OR NOT(P)
- NOT(P OR Q) IFF (NOT(P) AND NOT(Q))
- A IFF B, where A and B are logically equivalent

A propositional formula is unsatisfiable or a contradiction if all its truth table entries are false.

Examples:

- P AND NOT(P)
- NOT(A), where A is a tautology

A propositional formula is satisfiable if it is not unsatisfiable.

A truth assignment is a function from
a set of propositional variables to {T, F}.

Example $\tau : \{P, Q\} \to \{\text{T, F}\}$.

Each row in a truth table corresponds to a different truth assignment with the same domain.

An algorithm to determine if a propositional formula is satisfiable:
Construct its truth table.

If a formula has n variables, there are 2^n rows in its truth table.

SAT

The satisfiability problem or SAT:
decide whether a given propositional formula is satisfiable.
Input: a propositional formula
Output: YES if the formula is satisfiable, NO if it is unsatisfiable.

P = all decision problems that can be solved in polynomial time.

SAT

The satisfiability problem or SAT:

decide whether a given propositional formula is satisfiable.
Input: a propositional formula
Output: YES if the formula is satisfiable, NO if it is unsatisfiable.

P = all decision problems that can be solved in polynomial time.

NP = all decision problems that can be verified in polynomial time.

SAT $\in$ NP.

Example: (P OR NOT(Q)) AND (NOT(P) OR Q) is satisfiable,
because τ((P OR NOT(Q)) AND (NOT(P) OR Q)) = T,
where τ(P) = F and τ(Q) = F.

SAT

The satisfiability problem or SAT:

decide whether a given propositional formula is satisfiable.
Input: a propositional formula
Output: YES if the formula is satisfiable, NO if it is unsatisfiable.

P = all decision problems that can be solved in polynomial time.
NP = all decision problems that can be verified in polynomial time.

SAT $\in$ NP.

Example: (P OR NOT(Q)) AND (NOT(P) OR Q) is satisfiable,
because τ((P OR NOT(Q)) AND (NOT(P) OR Q)) = T,
where τ(P) = F and τ(Q) = F.

THEOREM[Cook] SAT $\in$ P if and only if P = NP.

DNF

A literal is a variable or the negation of a variable.

It is easy to solve satisfiability for a formula that is the conjunction of literals.

Examples:

- P AND NOT(Q) AND R is satisfiable.
- P AND NOT(Q) AND R AND NOT(P) is unsatisfiable.

DNF

A literal is a variable or the negation of a variable.

It is easy to solve satisfiability for a formula that is the conjunction of literals.

Examples:

- P AND NOT(Q) AND R is satisfiable.
- P AND NOT(Q) AND R AND NOT(P) is unsatisfiable.

Check that the formula does not contain a variable and its negation.

A propositional formula is in disjunctive normal form if it is a disjunction of conjunctions of literals.

DNF: a disjunction of conjunctions of literals

THEOREM Every propositional formula is logically equivalent to a propositional formula in DNF.

P	Q	R	
F	F	F	F
F	F	T	F
F	T	F	F
F	T	T	T
T	F	F	F
T	F	T	F
T	T	F	F
T	T	T	F

NOT(P) AND Q AND R

DNF: a disjunction of conjunctions of literals

THEOREM Every propositional formula is logically equivalent to a propositional formula in DNF.

P	Q	R		
F	F	F	F	
F	F	T	F	
F	T	F	F	
F	T	T	T	NOT(P) AND Q AND R
T	F	F	F	
T	F	T	T	P AND NOT(Q) AND R
T	T	F	F	
T	T	T	T	P AND Q AND R

DNF: a disjunction of conjunctions of literals

THEOREM Every propositional formula is logically equivalent to a propositional formula in DNF.

P	Q	R		
F	F	F	F	
F	F	T	F	
F	T	F	F	
F	T	T	T	NOT(P) AND Q AND R
T	F	F	F	
T	F	T	T	P AND NOT(Q) AND R
T	T	F	F	
T	T	T	T	P AND Q AND R

(NOT(P) AND Q AND R) OR (P AND NOT(Q) AND R) OR (P AND Q AND R) is a DNF formula for the last column.

DNF: a disjunction of conjunctions of literals

THEOREM Every propositional formula is logically equivalent to a propositional formula in DNF.

P	Q	R		
F	F	F	F	
F	F	T	F	
F	T	F	F	
F	T	T	T	NOT(P) AND Q AND R
T	F	F	F	
T	F	T	T	P AND NOT(Q) AND R
T	T	F	F	
T	T	T	T	P AND Q AND R

(NOT(P) AND Q AND R) OR (P AND R)
is another DNF formula for the last column.

CNF

A propositional formula is in conjunctive normal form
if it is a conjunction of disjunctions of literals.

A clause is a disjunction of literals,
so a CNF formula is a conjunction of clauses.

THEOREM Every propositional formula is logically equivalent to a propositional formula in CNF.

CNF: a conjunction of disjunctions of literals

THEOREM Every propositional formula is logically equivalent to a propositional formula in CNF.

P	Q	R	
F	F	F	T
F	F	T	F
F	T	F	T
F	T	T	T
T	F	F	F
T	F	T	T
T	T	F	T
T	T	T	T

CNF: a conjunction of disjunctions of literals

Construct the DNF for the complement,

P	Q	R			
F	F	F	T	F	
F	F	T	F	T	NOT(P) AND NOT(Q) AND R
F	T	F	T	F	
F	T	T	T	F	
T	F	F	F	T	P AND NOT(Q) AND NOT(R)
T	F	T	T	F	
T	T	F	T	F	
T	T	T	T	F	

[NOT(P) AND NOT(Q) AND R] OR
[P AND NOT(Q) AND NOT(R)]

CNF: a conjunction of disjunctions of literals

Construct the DNF for the complement, negate it,

NOT([NOT(P) AND NOT(Q) AND R] OR
[P AND NOT(Q) AND NOT(R)])

CNF: a conjunction of disjunctions of literals

Construct the DNF for the complement, negate it, apply DeMorgan's laws,

NOT([NOT(P) AND NOT(Q) AND R] OR
  [P AND NOT(Q) AND NOT(R)])
= NOT(NOT(P) AND NOT(Q) AND R) AND
  NOT(P AND NOT(Q) AND NOT(R))
= [NOT(NOT(P)) OR NOT(NOT(Q)) OR NOT(R)] AND
  [NOT(P) OR NOT(NOT(Q)) OR NOT(NOT(R))]

CNF: a conjunction of disjunctions of literals

Construct the DNF for the complement, negate it, apply DeMorgan's laws, and simplify.

NOT([NOT(P) AND NOT(Q) AND R] OR
    [P AND NOT(Q) AND NOT(R)])
= NOT(NOT(P) AND NOT(Q) AND R) AND
  NOT(P AND NOT(Q) AND NOT(R))
= [NOT(NOT(P)) OR NOT(NOT(Q)) OR NOT(R)] AND
  [NOT(P) OR NOT(NOT(Q)) OR NOT(NOT(R))]
= [P OR Q OR NOT(R)] AND [NOT(P) OR Q OR R]

CNF-SAT: decide whether a given propositional formula in CNF is satisfiable.

Input: a propositional formula in CNF

Output: YES if the formula is satisfiable, NO if it is unsatisfiable.

CNF-SAT is just as hard as SAT.

Lecture 4

Validity and Satisfiability of Predicate Logic Formulas

Prenex Normal Form

Predicate Logic Formulas

A predicate logic formula is an expression built up from predicate symbols, each of which has a fixed number of arguments, using connectives and universal and existential quantifiers.

The arguments of predicate symbols are variables and constants from specific domains.

$\forall x \in D.[P(x, y, 0) \text{ IMPLIES } \exists y \in D.(S(x, y) \text{ AND } G(y))]$

where $0 \in D$

$G : D \to \{\text{T, F}\}$

$S : D \times D \to \{\text{T, F}\}$

$P : D \times D \times D \to \{\text{T, F}\}$

A constant symbol denotes one particular element in a domain,
whereas a variable can be used to denote any element in a domain.

An occurrence of a variable x is quantified
if it occurs within a subformula of the form $\forall x \in D.E$ or $\exists x \in D.E$;
otherwise it is unquantified or free.
Constants are never quantified.

$\forall x \in D.[P(x, x, z) \text{ IMPLIES } \exists y \in D.(S(x, y) \text{ AND } G(y))]$

all occurrences of x and y are quantified
the occurrence of z is free

An occurrence of a variable x is quantified
if it occurs within a subformula of the form $\forall x \in D.E$ or $\exists x \in D.E$;
otherwise it is unquantified or free.

$\forall x \in D.[P(x, y, z) \text{ IMPLIES } \exists y \in D.(S(x, y) \text{ AND } G(y))]$

the first occurrence of y is free

Better:
$\forall x \in D.[P(x, y, z) \text{ IMPLIES } \exists w \in D.(S(x, w) \text{ AND } G(w))]$

DON'T use the same symbol for a free variable and a quantified variable in the same formula.

An occurrence of a variable x is quantified
if it occurs within a subformula of the form $\forall x \in D.E$ or $\exists x \in D.E$;
otherwise it is unquantified or free.

$\forall y \in D.[P(x,y,z) \text{ IMPLIES } \exists y \in D.(S(x,y) \text{ AND } G(y))]$
occurrences of x and z are free
the occurrences of y are quantified

Better:
$\forall y \in D.[P(x,y,z) \text{ IMPLIES } \exists w \in D.(S(x,w) \text{ AND } G(w))]$

DON'T use the same symbol for nested quantified variables.

The truth of a predicate logic formula depends on:
what the predicate symbols mean,
what the domains are, and
what the constant symbols refer to.

$p : D \to \{T, F\}$
$\forall x \in D . p(x)$

TRUE if $D = \{1, 3, 5\}$ and $p(x) =$ 'x is odd'
FALSE if $D = \{1, 2, 3\}$ and $p(x) =$ 'x is odd'
FALSE if $D = \{1, 3, 5\}$ and $p(x) =$ 'x is even'

The truth of a predicate logic formula depends on:
what the predicate symbols mean,
what the domains are, and
what the constant symbols refer to.

$f : D \times D \to D$
$id \in D$

$(\forall x \in D.f(x, id) = x)$ IMPLIES $(\forall x \in D.\exists y \in D.f(x, y) = id)$

FALSE if $D = \mathbb{Z}$, $f(x, y) =$ '$x \times y$', and $id = 1$
TRUE if $D = \mathbb{Z}$, $f(x, y) =$ '$x + y$', and $id = 0$

Interpretation

An interpretation of a predicate logic formula with no free variables consists of

- a nonempty set for each domain in the formula,
- a function from the relevant domain to the relevant range for each function symbol in the formula, and
- a domain element for each constant symbol.

In particular, each predicate symbol in the formula must have a function from the relevant domain to $\{\mathrm{T}, \mathrm{F}\}$.

If $D = \phi$, then
$\forall x \in D.p(x)$ is always vacuously true
$\exists x \in D.p(x)$ is always false.

Interpretation

An interpretation of a predicate logic formula consists of:

- a nonempty set for each domain in the formula,
- a function from the relevant domain to the relevant range for each function symbol in the formula,
- a domain element for each constant symbol, and
- an element of the relevant domain for each free variable.

A valuation maps each free variable to a domain element.

$\forall x \in D.\exists y \in D.(q(x, y) \text{ AND } r(x, y, z))$

TRUE if $D = \mathbb{N}$, $q(x, y) =$ '$x \geq y$', $r(x, y, z) =$ '$x \cdot y = z$', $z = 0$
FALSE if $D = \mathbb{N}$, $q(x, y) =$ '$x \geq y$', $r(x, y, z) =$ '$x \cdot y = z$', $z = 1$

Validity and Satisfiability

A predicate logic formula is valid or a tautology if it true for all interpretations.

A predicate logic formula is satisfiable if it is true for some interpretation.

A predicate logic formula is unsatisfiable if it is false for all interpretations.

$(\forall x \in D.a(x))$ IMPLIES $\exists x \in D.a(x)$ VALID
$(\forall x \in D.a(x))$ IMPLIES $a(c)$ VALID
$(\forall x \in D.a(x))$ IMPLIES $a(y)$ VALID
$(\exists x \in D.a(x))$ IMPLIES $\forall x \in D.a(x)$ SATISFIABLE, not VALID

employee	gender	salary
Andy	M	0
Donna	F	3000
Leslie	F	5000
Ron	M	5700
Tom	M	1800

$(\exists x \in D.a(x))$ IMPLIES $\forall x \in D.a(x)$

FALSE if $D = E$ and $a(x) =$ 'employee x is female'.
TRUE if $D = E$ and $a(x) =$ 'employee x made more than 6000'.
TRUE if $|D| = 1$.

employee	gender	salary
Andy	M	0
Donna	F	3000
Leslie	F	5000
Ron	M	5700
Tom	M	1800

$\exists y \in D . \forall x \in D . q(x, y)$

Consider interpretations in which $D = E$.

TRUE if $q(x, y) =$ ‘employee x made no less than employee y’.
FALSE if $q(x, y) =$ ‘employees x and y have the same gender’.

employee	gender	salary
Andy	M	0
Donna	F	3000
Leslie	F	5000
Ron	M	5700
Tom	M	1800

$\forall x \in D. \exists y \in D. q(x, y)$

employee	gender	salary
Andy	M	0
Donna	F	3000
Leslie	F	5000
Ron	M	5700
Tom	M	1800

$\forall x \in D . \exists y \in D . q(x, y)$

Consider interpretations in which $D = E$.

TRUE if $q(x, y) =$ 'employee x made no less than employee y'.
TRUE if $q(x, y) =$ 'employees x and y have the same gender'.
FALSE if $q(x, y) =$ 'employee x made less than employee y'.

Logical Implication and Equivalence

E logically implies E' means that E' is true in every interpretation that makes E true.
E and E' are logically equivalent if E logically implies E' and E' logically implies E.

$\exists x \in D.(p(x) \text{ OR } q(x))$ is logically equivalent to
$(\exists x \in D.p(x)) \text{ OR } (\exists x \in D.q(x))$

$\exists x \in D.(p(x) \text{ AND } q(x))$ logically implies that
$(\exists x \in D.p(x)) \text{ AND } (\exists x \in D.q(x))$

If x does not occur in E, then
$\exists x \in D.(E \text{ AND } q(x))$ is logically equivalent to
$E \text{ AND } (\exists x \in D.q(x))$.

Logical Implication and Equivalence

E logically implies E' means that E' is true in every interpretation that makes E true.
E and E' are logically equivalent if E logically implies E' and E' logically implies E.

$\forall x \in D.(p(x) \text{ AND } q(x))$ is logically equivalent to
$(\forall x \in D.p(x)) \text{ AND } (\forall x \in D.q(x))$

$(\forall x \in D.p(x)) \text{ OR } (\forall x \in D.q(x))$ logically implies that
$\forall x \in D.(p(x) \text{ OR } q(x))$

If x does not occur in E then
$\forall x \in D.(E \text{ OR } q(x))$ is logically equivalent to
$E \text{ OR } (\forall x \in D.q(x))$.

Logical Implication and Equivalence

If x does not occur in E, then

$\exists x \in D.(E \text{ IMPLIES } q(x))$ is logically equivalent to
$E \text{ IMPLIES } \exists x \in D.q(x)$.

$\forall x \in D.(E \text{ IMPLIES } q(x))$ is logically equivalent to
$E \text{ IMPLIES } \forall x \in D.q(x)$.

$\exists x \in D.(p(x) \text{ IMPLIES } E)$ is logically equivalent to
$(\forall x \in D.p(x)) \text{ IMPLIES } E$.

$\forall x \in D.(p(x) \text{ IMPLIES } E)$ is logically equivalent to
$(\exists x \in D.p(x)) \text{ IMPLIES } E$.

Prenex Normal Form

A predicate logic formula is in prenex normal form if and only if it is of the form

$$Q_1 x_1 \in D_1 . Q_2 x_2 \in D_2 . \cdots Q_k x_k \in D_k . E(x_1, \ldots, x_k),$$

where $E(x_1, \ldots, x_k)$ is a formula without quantifiers and,
for all $i = 1, \ldots, k$, Q_i is either $\forall$ or $\exists$

Any predicate logic formula can be converted to prenex normal form by applying a sequence of transformations:

NOT($\exists x \in D.p(x)$) can be transformed to $\forall x \in D$.NOT($p(x)$)

NOT ($\forall x \in D.p(x)$) can be transformed to $\exists x \in D$.NOT($p(x)$)

Prenex Normal Form

$(\exists x \in D.p(x))$ OR $(\exists x \in D.q(x))$ can be transformed to $\exists x \in D.(p(x)$ OR $q(x))$

$(\forall x \in D.p(x))$ AND $(\forall x \in D.q(x))$ can be transformed to $\forall x \in D.(p(x)$ AND $q(x))$

$(\exists x \in D.p(x))$ AND $(\exists x \in D.q(x))$ CANNOT be transformed to $\exists x \in D.(p(x)$ AND $q(x))$

$(\forall x \in D.p(x))$ OR $(\forall x \in D.q(x))$ CANNOT be transformed to $\forall x \in D.(p(x)$ OR $q(x))$

Prenex Normal Form

If x does not occur in E, then

E AND $\exists x \in D.q(x)$ can be transformed to $\exists x \in D.(E \text{ AND } q(x))$

$(\exists x \in D.p(x))$ AND E can be transformed to $\exists x \in D.(p(x) \text{ AND } E)$

E AND $\forall x \in D.q(x)$ can be transformed to $\forall x \in D.(E \text{ AND } q(x))$

$(\forall x \in D.p(x))$ AND E can be transformed to $\forall x \in D.(p(x) \text{ AND } E)$

Prenex Normal Form

If x does not occur in E, then

E OR $\exists x \in D.q(x)$ can be transformed to $\exists x \in D.(E \text{ OR } q(x))$

$(\exists x \in D.p(x))$ OR E can be transformed to $\exists x \in D.(p(x) \text{ OR } E)$

E OR $\forall x \in D.q(x)$ can be transformed to $\forall x \in D.(E \text{ OR } q(x))$

$(\forall x \in D.p(x))$ OR E can be transformed to $\forall x \in D.(p(x) \text{ OR } E)$

If x does not occur in E, then

E IMPLIES $\exists x \in D.q(x)$ can be transformed to $\exists x \in D.(E \text{ IMPLIES } q(x))$

$(\exists x \in D.p(x))$ IMPLIES E can be transformed to $\forall x \in D.(p(x) \text{ IMPLIES } E)$

E IMPLIES $\forall x \in D.q(x)$ can be transformed to $\forall x \in D.(E \text{ IMPLIES } q(x))$

$(\forall x \in D.p(x))$ IMPLIES E can be transformed to $\exists x \in D.(p(x) \text{ IMPLIES } E)$

Example:

$(f(x, c) \text{ OR } \exists y \in A.g(y)) \text{ IMPLIES } \forall z \in B.h(z)$
is transformed to
$(\exists y \in A.(f(x, c) \text{ OR } g(y))) \text{ IMPLIES } \forall z \in B.h(z)$
is transformed to
$\forall y \in A.[(f(x, c) \text{ OR } g(y)) \text{ IMPLIES } \forall z \in B.h(z)]$
is transformed to
$\forall y \in A.\forall z \in B.[(f(x, c) \text{ OR } g(y)) \text{ IMPLIES } h(z)]$

Another logically equivalent formula in prenex normal form:
$\forall z \in B.\forall y \in A.[\text{NOT}(f(x, c) \text{ OR } g(y)) \text{ OR } h(z)]$

Problem 1

Determine whether each of these propositional formulas is valid, satisfiable but not valid, or unsatisfiable:

1. (P IMPLIES Q) IMPLIES P
2. P IMPLIES (Q IMPLIES P)

Problem 1

Determine whether each of these propositional formulas is valid, satisfiable but not valid, or unsatisfiable:

1. $(P$ IMPLIES $Q)$ IMPLIES P
   This formula is satisfiable, but not valid.
   The truth assignment $P = Q = \mathrm{T}$ makes it true.
   The truth assignment $P = Q = \mathrm{F}$ makes it false.
2. P IMPLIES $(Q$ IMPLIES $P)$

Problem 1

Determine whether each of these propositional formulas is valid, satisfiable but not valid, or unsatisfiable:

1. (P IMPLIES Q) IMPLIES P
2. P IMPLIES (Q IMPLIES P)

   if $P = \text{F}$, then the formula is vacuously true.

   if $P = \text{T}$, then Q IMPLIES P is true, so the formula is true.

   Therefore, this formula is valid.

Problem 2

Consider the truth table for the Boolean predicate $M(P, Q, R)$, which is true when exactly two of P, Q, and R are true.

P	Q	R	M
T	T	T	F
T	T	F	T
T	F	T	T
T	F	F	F
F	T	T	T
F	T	F	F
F	F	T	F
F	F	F	F

1. Give a propositional formula in DNF for $M(P, Q, R)$.
2. Give a propositional formula in CNF for $M(P, Q, R)$.

$M(P,Q,R)$ is true when exactly two of P, Q, and R are true.

P	Q	R	M	
T	T	T	F	
T	T	F	T	P AND Q AND (NOT R)
T	F	T	T	P AND (NOT Q) AND R
T	F	F	F	
F	T	T	T	(NOT P) AND Q AND R
F	T	F	F	
F	F	T	F	
F	F	F	F	

1. Give a propositional formula in DNF for $M(P,Q,R)$.

(P AND Q AND (NOT R)) OR
(P AND (NOT Q) AND R) OR
((NOT P) AND Q AND R)

$M(P, Q, R)$ is true when exactly two of P, Q, and R are true.

P	Q	R	M
T	T	T	F
T	T	F	T
T	F	T	T
T	F	F	F
F	T	T	T
F	T	F	F
F	F	T	F
F	F	F	F

2. Give a propositional formula in CNF for $M(P, Q, R)$.

(P OR Q) AND (P OR R) AND (Q OR R)
AND ((NOT P) OR (NOT Q) OR (NOT R))

At least two of P, Q, and R are true.

At least one of P, Q, and R is false.

$M(P, Q, R)$ is true when exactly two of P, Q, and R are true.

P	Q	R	M	
T	T	T	F	P AND Q AND R
T	T	F	T	
T	F	T	T	
T	F	F	F	P AND (NOT Q) AND (NOT R)
F	T	T	T	
F	T	F	F	(NOT P) AND Q AND (NOT R)
F	F	T	F	(NOT P) AND (NOT Q) AND R
F	F	F	F	(NOT P) AND (NOT Q) AND (NOT R)

Another approach.
NOT ((P AND Q AND R) OR (P AND (NOT Q) AND (NOT R)) OR
((NOT P) AND Q AND (NOT R)) OR
((NOT P) AND (NOT Q) AND R) OR
((NOT P) AND (NOT Q) AND (NOT R)))

2. Give a propositional formula in CNF for $M(P, Q, R)$.

NOT ((P AND Q AND R) OR
(P AND (NOT Q) AND (NOT R)) OR
((NOT P) AND Q AND (NOT R)) OR
((NOT P) AND (NOT Q) AND R) OR
((NOT P) AND (NOT Q) AND (NOT R)))

$=$ ((NOT P) OR (NOT Q) OR (NOT R)) AND
((NOT P) OR Q OR R) AND
(P OR (NOT Q) OR R) AND
(P OR Q OR (NOT R)) AND
(P OR Q OR R)

You can see that this answer can be transformed into the first answer.

((NOT P) OR (NOT Q) OR (NOT R)) AND
((NOT P) OR Q OR R) AND
(P OR (NOT Q) OR R) AND
(P OR Q OR (NOT R)) AND
(P OR Q OR R)

= ((NOT P) OR (NOT Q) OR (NOT R)) AND
((NOT P) OR Q OR R) AND (P OR Q OR R) AND
(P OR (NOT Q) OR R) AND (P OR Q OR R) AND
(P OR Q OR (NOT R)) AND (P OR Q OR R)

= ((NOT P) OR (NOT Q) OR (NOT R)) AND
(Q OR R) AND
(P OR R) AND
(P OR Q)

Problem 3

Consider a Boolean circuit with four Boolean inputs, x_0, x_1, y_0, y_1 and three Boolean outputs z_0, z_1, and e,
where x_1x_0, y_1y_0, and z_1z_0 are the binary representations of the numbers x, y, and z, respectively.

If $y \neq 0$, then $e = 0$ and z is the quotient of x divided by y.
If $y = 0$, then $e = 1$ and $z_1 = z_0 = 0$.
The bit e is an error indicator.

For each of the three bits, e, z_1 and z_0, write a propositional formula in DNF or CNF (with variables x_0, x_1, y_0 and y_1) which expresses the value of that bit.

Problem 3

$e = \text{NOT } (y_1) \text{ AND NOT } (y_0)$

$x = x_1 x_0 \quad y = y_1 y_0 \quad z = z_1 z_0$

$$z = \begin{cases} 0 & \text{if } y = 0 \\ \lfloor x/y \rfloor & \text{if } y \neq 0 \end{cases}$$

The most significant bit of the quotient, z_1, is 1 if and only if $z \geq 2$.

Since $x \leq 3$, the quotient of x divided by y is at most 1 if $y \geq 2$.
Thus $z \geq 2$ if and only if $y = 1$ and $x \geq 2$.
Hence $z_1 = y_0 \text{ AND NOT } (y_1) \text{ AND } x_1$.

These two formulas are in both DNF and CNF.

Problem 3

$x = x_1x_0 \quad y = y_1y_0 \quad z = z_1z_0$

$$z = \begin{cases} 0 & \text{if } y = 0 \\ \lfloor x/y \rfloor & \text{if } y \neq 0 \end{cases}$$

If $y = 1$ (i.e. $y_1 = 0$ and $y_0 = 1$), then $z_0 = x_0$.

If $x < y$ then $z = 0$, so $z_0 = 0$.

Otherwise $2 \leq y \leq x \leq 3$. In this case, $z = 1$, so $z_0 = 1$.

Note that if $2 \leq y \leq 3$, then $y_1 = 1 = x_1$.

and either $x_0 = 1$ or $y_0 = 0$.

Otherwise $2 \leq y \leq x \leq 3$. In this case, $z = 1$, so $z_0 = 1$.

Note that $y_1 = 1 = x_1$, so

it is not the case that $x_0 = 0$ and $y_0 = 1$.

Hence, either $x_0 = 1$ or $y_0 = 0$.

Thus $z_0 =$ (NOT (y_1) AND y_0 AND x_0) OR

(y_1 AND x_1 AND [x_0 OR NOT (y_0)]).

However, this formula is not in DNF or CNF.

Problem 3

$$z_0 = (\text{NOT } (y_1) \text{ AND } y_0 \text{ AND } x_0)$$
$$\quad \text{OR } (y_1 \text{ AND } x_1 \text{ AND } [x_0 \text{ OR NOT } (y_0)]).$$

By distributivity,

$$y_1 \text{ AND } x_1 \text{ AND } [x_0 \text{ OR NOT } (y_0)]$$
$$= (y_1 \text{ AND } x_1 \text{ AND } x_0) \text{ OR } (y_1 \text{ AND } x_1 \text{ AND NOT } (y_0)).$$

Hence $z_0 = (\text{NOT } (y_1) \text{ AND } y_0 \text{ AND } x_0) \text{ OR }$
$(y_1 \text{ AND } x_1 \text{ AND } x_0) \text{ OR } (y_1 \text{ AND } x_1 \text{ AND NOT } (y_0))$,

which is in DNF.

Problem 3

A more tedious way to solve this problem is to use a truth table:

x_1	x_0	y_1	y_0	z_1	z_0
T	T	T	T	F	T
T	T	T	F	F	T
T	T	F	T	T	T
T	T	F	F	F	F
T	F	T	T	F	F
T	F	T	F	F	T
T	F	F	T	T	F
T	F	F	F	F	F
F	T	T	T	F	F
F	T	T	F	F	T
F	T	F	T	F	T
F	T	F	F	F	F
F	F	T	T	F	F
F	F	T	F	F	F
F	F	F	T	F	F
F	F	F	F	F	F

$z_1 =$
(x_1 AND NOT (x_0) AND NOT (y_1) AND y_0)
OR
(x_1 AND x_0 AND NOT (y_1) AND y_0)

Problem 3

A more tedious way to solve this problem is to use a truth table:

x_1	x_0	y_1	y_0	z_1	z_0
T	T	T	T	F	T
T	T	T	F	F	T
T	T	F	T	T	T
T	T	F	F	F	F
T	F	T	T	F	F
T	F	T	F	F	T
T	F	F	T	T	F
T	F	F	F	F	F
F	T	T	T	F	F
F	T	T	F	F	F
F	T	F	T	F	T
F	T	F	F	F	F
F	F	T	T	F	F
F	F	T	F	F	F
F	F	F	T	F	F
F	F	F	F	F	F

$z_0 =$
(NOT (x_1) AND x_0 AND NOT (y_1) AND y_0)
OR
(x_1 AND NOT (x_0) AND y_1 AND NOT (y_0))
OR
(x_1 AND x_0 AND NOT (y_1) AND y_0)
OR
(x_1 AND x_0 AND y_1 AND NOT (y_0))
OR
(x_1 AND x_0 AND y_1 AND y_0)

Problem 4

Determine the free occurrences of each variable in the predicate logic formula

$$[\forall x \in D. \exists y \in D. (R(x, y, z) \text{ IMPLIES } \exists w \in D. Q(w, x, y))]$$
$$\text{IMPLIES } \forall y \in D. Q(w, w, y)$$

Problem 4

Determine the free occurrences of each variable in the predicate logic formula

$$[\forall x \in D. \exists y \in D. (R(x, y, z) \text{ IMPLIES } \exists w \in D. Q(w, x, y))]$$
$$\text{IMPLIES } \forall y \in D. Q(w, w, y)$$

Problem 5

Translate the formula

$$\forall x \in D. \forall y \in D. [(G(y) \text{ AND } \forall y \in D.(S(x,y) \text{ IMPLIES } B(y))) \text{ IMPLIES NOT } (S(y,x))]$$

into an English sentence, using an interpretation where D is a nonempty set of people,
$G(y)$ = "person y is a girl"
$B(y)$ = "person y is a boy" and
$S(x,y)$ = "person x is a sibling of person y".

Problem 5

Translate the formula

$$\forall x \in D.\forall y \in D.[(G(y) \text{ AND } \forall y \in D.(S(x,y) \text{ IMPLIES } B(y))) \text{ IMPLIES NOT } (S(y,x))]$$

into an English sentence, using an interpretation where D is a nonempty set of people,
$G(y) =$ "person y is a girl"
$B(y) =$ "person y is a boy" and
$S(x,y) =$ "person x is a sibling of person y".

$G(y)$ and $S(y,x)$ are bound to the first $\forall y \in D$
$S(x,y)$ and $B(y)$ are bound to the second $\forall y \in D$
If a variable is in the scope of multiple quantifiers, it is bound to the innermost quantifier applied to that variable.

$$\forall x \in D.\forall y \in D.[(G(y) \text{ AND } \forall z \in D.(S(x,z) \text{ IMPLIES } B(z))) \text{ IMPLIES NOT } (S(y,x))]$$

Problem 5

$$\forall x \in D.\forall y \in D.[(G(y) \text{ AND } \forall z \in D.(S(x,z) \text{ IMPLIES } B(z)))$$
$$\text{IMPLIES NOT } (S(y,x))]$$

The literal translation:

For all people x and y in D, if y is a girl and every sibling of x is a boy, then y is not a sibling of x.

Other correct translations:

Every girl (x) is not the sibling of anyone (y) whose siblings are all boys.

Every girl is not the sister of anyone who only has brothers.

This follows from the following facts:

If a girl x is a sibling of y, then x is the sister of y.

If a sibling z is a boy, then z is a brother.

Everyone who only has brothers has no sisters.

Problem 5

$$\forall x \in D.\forall y \in D.[(G(y) \text{ AND } \forall z \in D.(S(x,z) \text{ IMPLIES } B(z))) \text{ IMPLIES NOT } (S(y,x))]$$

The translation:

If a person has only brothers, they have no sisters.

is not as good, because it does not make the universal quantification clear. The universal quantification is indicated by the word "a" and might be misunderstood by some people.

A similar problem applies to the translation:

If there is a girl and a person whose siblings are all boys, they are not siblings.

Problem 5

$$\forall x \in D. \forall y \in D. [(G(y) \text{ AND } \forall z \in D.(S(x,z) \text{ IMPLIES } B(z))) \text{ IMPLIES NOT } (S(y,x))]$$

An incorrect translation: Girls don't have brothers in this set D.

Here is why this is incorrect. Consider the set $D = \{b, g, g'\}$, where g and g' are girls, b is a boy, and all three are siblings of one another.

Note that every person $x \in D$ has a sibling who is not a boy, so $\forall z \in D.(S(x,z) \text{ IMPLIES } B(z))$ is false. Therefore $(G(y) \text{ AND } \forall z \in D.(S(x,z) \text{ IMPLIES } B(z))) \text{ IMPLIES NOT } (S(y,x))$ is true. Hence the formula is true.

However, the English sentence is false, because both g and g' have b as their brother.

Problem 5

$$\forall x \in D.\forall y \in D.[(G(y) \text{ AND } \forall z \in D.(S(x,z) \text{ IMPLIES } B(z))) \text{ IMPLIES NOT } (S(y,x))]$$

The same counterexample applies to the incorrect translation:

Every girl is not a sibling of anyone who has a brother.

Consider the set $D = \{b, g, g'\}$, where g and g' are girls, b is a boy, and all three are siblings of one another.

The English sentence is false, because g is a sibling of g' and b is a brother of g'.

Problem 5

$$\forall x \in D.\forall y \in D.[(G(y) \text{ AND } \forall z \in D.(S(x,z) \text{ IMPLIES } B(z)))$$
$$\text{IMPLIES NOT } (S(y,x))]$$

More generally, if the sets B and G are disjoint, the formula is true. Consider any people x and y such that $G(y)$ AND $\forall z \in D.(S(x,z)$ IMPLIES $B(z))$ is true. Then y is a girl and if x is a sibling of y, then y is a boy and, hence, not a girl. So this means that x is not a sibling of y and therefore y is not a sibling of x. Hence NOT $S(y,x)$ is true. ∎

Problem 5

$$\forall x \in D.\forall y \in D.[(G(y) \text{ AND } \forall z \in D.(S(x,z) \text{ IMPLIES } B(z))) \text{ IMPLIES NOT } (S(y,x))]$$

Another incorrect translation: If everyone is a girl and everyone with a sibling is a boy then no one has a sibling.

Consider the set $D = \{a, b\}$, where a is both a girl and a boy, b is a boy, and a and b are siblings.

Since $a \in D$ is a girl and all siblings of a are boys, $G(a)$ AND $\forall z \in D.(S(a,z) \text{ IMPLIES } B(z))$ is true.

But a and b are siblings, so $S(a, b)$ is true. Thus the formula is false.

However, the English sentence is vacuously true. Because b is not a girl, everyone is a girl is false.

Problem 6

Consider the predicate logic formula

$$\forall A \in S.\forall B \in S.\forall C \in S.\forall D \in S.[(A \times B \subseteq C \times D) \text{ IMPLIES } ((A \subseteq C) \text{ AND } (B \subseteq D))].$$

1. Give an interpretation which makes this formula true, where $\times$ denotes the Cartesian product of sets and $\subseteq$ denotes subset.
2. Give an interpretation which makes this formula false, where $\times$ denotes the Cartesian product of sets and $\subseteq$ denotes subset.

Consider the predicate logic formula

$$\forall A \in S. \forall B \in S. \forall C \in S. \forall D \in S. [(A \times B \subseteq C \times D) \text{ IMPLIES } ((A \subseteq C) \text{ AND } (B \subseteq D))]$$

1. Give an interpretation which makes this formula true, where $\times$ denotes the Cartesian product of sets and $\subseteq$ denotes subset.

Let $S = \{ \{s\} \}$ consist of one set,
which contains a single element s.
The only choices for A, B, C, and D are $A = B = C = D = \{s\}$.
Since $A \times B = \{(s, s)\} = C \times D$,
both the hypothesis and the conclusion are true.
Hence the implication is true.
Therefore the formula is true under this interpretation.

Consider the predicate logic formula

$$\forall A \in S\ \forall B \in S\ \forall C \in S\ \forall D \in S\ [(A \times B \subseteq C \times D) \text{ IMPLIES } ((A \subseteq C) \text{ AND } (B \subseteq D))].$$

2. Give an interpretation which makes this formula false, where $\times$ denotes the Cartesian product of sets and $\subseteq$ denotes subset.

Let $S = \{\ \phi,\ \{r\},\ \{s\}\ \}$.
Let $A = \phi$, $B = \{r\}$, and $C = D = \{s\}$.
Since $A \times B = \phi \subseteq \{(s, s)\} = C \times D$, the hypothesis is true.
But $B \not\subseteq D$, so the conclusion is false.
Hence the implication is false and the formula is false under this interpretation.

Problem 1

Consider the predicate logic formula

$$\exists y \in S.[(\forall x \in S.A(x)) \text{ IMPLIES } B(y)] \text{ IFF}$$
$$\exists y \in S.[\forall x \in S.(A(x) \text{ IMPLIES } B(y))]$$

1. What is the smallest possible size for the domain S in an interpretation so that the formula is true?
2. What is the smallest possible size for the domain S in an interpretation so that the formula is false?

Justify your answers.

Consider the predicate logic formula

$$\exists y \in S.[(\forall x \in S.A(x)) \text{ IMPLIES } B(y)] \text{ IFF}$$
$$\exists y \in S.[\forall x \in S.(A(x) \text{ IMPLIES } B(y))].$$

1. What is the smallest possible size for the domain S in an interpretation so that the formula is true?

In any interpretation, $|S| \geq 1$.
Suppose that $|S| = 1$. Let c be the only element of S.
Then $\forall x \in S.A(x)$ is logically equivalent to $A(c)$, so $(\forall x \in S.A(x))$ IMPLIES $B(y)$ is logically equivalent to $A(c)$ IMPLIES $B(y)$.

Also, $\forall x \in S.(A(x) \text{ IMPLIES } B(y))$ is logically equivalent to $A(c)$ IMPLIES $B(y)$.
Hence $\exists y \in S.[(\forall x \in S.A(x)) \text{ IMPLIES } B(y)]$ is logically equivalent to $\exists y \in S.[\forall x \in S.(A(x) \text{ IMPLIES } B(y))]$.
Thus, every interpretation with $|S| = 1$ makes the formula true.

Consider the predicate logic formula

$$\exists y \in S.[(\forall x \in S.A(x)) \text{ IMPLIES } B(y)] \text{ IFF}$$
$$\exists y \in S.[\forall x \in S.(A(x) \text{ IMPLIES } B(y))].$$

2. What is the smallest possible size for the domain S in an interpretation so that the formula is false?

Consider the interpretation in which $S = \{0, 1\}$, $A(0) = F$, $A(1) = T$, and $B(0) = B(1) = F$.

Then $\forall x \in S.A(x)$ is false, so for every $y \in \{0, 1\}$, $(\forall x \in S.A(x))$ IMPLIES $B(y)$.
Thus $\exists y \in S.[(\forall x \in S.A(x))$ IMPLIES $B(y)]$ is true.
However, since $A(1) = T$ and $B(0) = F$, $A(1)$ IMPLIES $B(0)$ is F and, hence $\forall x \in S.(A(x)$ IMPLIES $B(0))$ is F.

Similarly, $\forall x \in S.(A(x)$ IMPLIES $B(1))$ is F.
Thus $y \in S.[\forall x \in S.(A(x)$ IMPLIES $B(y))]$ is F.
Hence, the formula is false under this interpretation and the smallest possible size for S is 2.

Problem 2

Translate the formula

$$[\exists x \in \mathbb{N}.(x = y)] \text{ IMPLIES } \exists x \in \mathbb{N}.[x = 0 \text{ OR NOT } (\exists y \in \mathbb{N}.(y < 0))]$$

into a logically equivalent formula in Prenex Normal Form.
Use brackets where necessary to avoid ambiguity.

Problem 2

Translate the formula

$$[\exists x \in \mathbb{N}.(x = y)] \text{ IMPLIES } \exists x \in \mathbb{N}.[x = 0 \text{ OR NOT } (\exists y \in \mathbb{N}.(y < 0))]$$

into a logically equivalent formula in Prenex Normal Form.
Use brackets where necessary to avoid ambiguity.
Substitute z for the second quantified x and w for the quantified y.

$$[\exists x \in \mathbb{N}.(x = y)] \text{ IMPLIES } \exists z \in \mathbb{N}.(z = 0 \text{ OR NOT } (\exists w \in \mathbb{N}.(w < 0)))$$

Problem 2

Translate the formula

$$[\exists x \in \mathbb{N}.(x = y)] \text{ IMPLIES } \exists x \in \mathbb{N}.[x = 0 \text{ OR NOT } (\exists y \in \mathbb{N}.(y < 0))]$$

into a logically equivalent formula in Prenex Normal Form.
Use brackets where necessary to avoid ambiguity.
Substitute z for the second quantified x and w for the quantified y.

$$[\exists x \in \mathbb{N}.(x = y)] \text{ IMPLIES } \exists z \in \mathbb{N}.[z = 0 \text{ OR NOT } (\exists w \in \mathbb{N}.(w < 0)))$$

Then apply a sequence of transformations to get Prenex Normal Form.

Apply a sequence of transformations

$[\exists x \in \mathbb{N}.(x = y)]$ IMPLIES $\exists z \in \mathbb{N}.(z = 0$ OR NOT $(\exists w \in \mathbb{N}.(w < 0)))$

$[\exists x \in \mathbb{N}.(x = y)]$ IMPLIES $\exists z \in \mathbb{N}.(z = 0$ OR $\forall w \in \mathbb{N}.$NOT $(w < 0))$

Apply a sequence of transformations

$$[\exists x \in \mathbb{N}.(x = y)] \text{ IMPLIES } \exists z \in \mathbb{N}.(z = 0 \text{ OR NOT } (\exists w \in \mathbb{N}.(w < 0)))$$

$$[\exists x \in \mathbb{N}.(x = y)] \text{ IMPLIES } \exists z \in \mathbb{N}.(z = 0 \text{ OR } \forall w \in \mathbb{N}.\text{NOT } (w < 0))$$

$$[\exists x \in \mathbb{N}.(x = y)] \text{ IMPLIES } \exists z \in \mathbb{N}.\forall w \in \mathbb{N}.(z = 0 \text{ OR NOT } (w < 0))$$

Apply a sequence of transformations

$$[\exists x \in \mathbb{N}.(x = y)] \text{ IMPLIES } \exists z \in \mathbb{N}.(z = 0 \text{ OR NOT } (\exists w \in \mathbb{N}.(w < 0)))$$

$$[\exists x \in \mathbb{N}.(x = y)] \text{ IMPLIES } \exists z \in \mathbb{N}.(z = 0 \text{ OR } \forall w \in \mathbb{N}.\text{NOT } (w < 0))$$

$$[\exists x \in \mathbb{N}.(x = y)] \text{ IMPLIES } \exists z \in \mathbb{N}.\forall w \in \mathbb{N}.(z = 0 \text{ OR NOT } (w < 0))$$

$$\forall x \in \mathbb{N}.[(x = y) \text{ IMPLIES } \exists z \in \mathbb{N}.\forall w \in \mathbb{N}.(z = 0 \text{ OR NOT } (w < 0))]$$

Apply a sequence of transformations

$$[\exists x \in \mathbb{N}.(x = y)] \text{ IMPLIES } \exists z \in \mathbb{N}.(z = 0 \text{ OR NOT } (\exists w \in \mathbb{N}.(w < 0)))$$

$$[\exists x \in \mathbb{N}.(x = y)] \text{ IMPLIES } \exists z \in \mathbb{N}.(z = 0 \text{ OR } \forall w \in \mathbb{N}.\text{NOT } (w < 0))$$

$$[\exists x \in \mathbb{N}.(x = y)] \text{ IMPLIES } \exists z \in \mathbb{N}.\forall w \in \mathbb{N}.(z = 0 \text{ OR NOT } (w < 0))$$

$$\forall x \in \mathbb{N}.[(x = y) \text{ IMPLIES } \exists z \in \mathbb{N}.\forall w \in \mathbb{N}.(z = 0 \text{ OR NOT } (w < 0))]$$

$$\forall x \in \mathbb{N}.\exists z \in \mathbb{N}.[(x = y) \text{ IMPLIES } \forall w \in \mathbb{N}.(z = 0 \text{ OR NOT } (w < 0))]$$

Apply a sequence of transformations

$$[\exists x \in \mathbb{N}.(x = y)] \text{ IMPLIES } \exists z \in \mathbb{N}.(z = 0 \text{ OR NOT } (\exists w \in \mathbb{N}.(w < 0)))$$

$$[\exists x \in \mathbb{N}.(x = y)] \text{ IMPLIES } \exists z \in \mathbb{N}.(z = 0 \text{ OR } \forall w \in \mathbb{N}.\text{NOT } (w < 0))$$

$$[\exists x \in \mathbb{N}.(x = y)] \text{ IMPLIES } \exists z \in \mathbb{N}.\forall w \in \mathbb{N}.(z = 0 \text{ OR NOT } (w < 0))$$

$$\forall x \in \mathbb{N}.[(x = y) \text{ IMPLIES } \exists z \in \mathbb{N}.\forall w \in \mathbb{N}.(z = 0 \text{ OR NOT } (w < 0))]$$

$$\forall x \in \mathbb{N}.\exists z \in \mathbb{N}.[(x = y) \text{ IMPLIES } \forall w \in \mathbb{N}.(z = 0 \text{ OR NOT } (w < 0))]$$

$$\forall x \in \mathbb{N}.\exists z \in \mathbb{N}.\forall w \in \mathbb{N}.[(x = y) \text{ IMPLIES } (z = 0 \text{ OR NOT } (w < 0))]$$

Apply an alternative sequence of transformations

$[\exists x \in \mathbb{N}.(x = y)]$ IMPLIES $\exists z \in \mathbb{N}.(z = 0$ OR NOT $(\exists w \in \mathbb{N}.(w < 0)))$

$[\exists x \in \mathbb{N}.(x = y)]$ IMPLIES $\exists z \in \mathbb{N}.(z = 0$ OR $\forall w \in \mathbb{N}.$NOT $(w < 0))$

$[\exists x \in \mathbb{N}.(x = y)]$ IMPLIES $\exists z \in \mathbb{N}.\forall w \in \mathbb{N}.(z = 0$ OR NOT $(w < 0))$

NOT $[\exists x \in \mathbb{N}.(x = y)]$ OR $\exists z \in \mathbb{N}.\forall w \in \mathbb{N}.(z = 0$ OR NOT $(w < 0))$

Apply an alternative sequence of transformations

$$[\exists x \in \mathbb{N}.(x = y)] \text{ IMPLIES } \exists z \in \mathbb{N}.(z = 0 \text{ OR NOT } (\exists w \in \mathbb{N}.(w < 0)))$$

$$[\exists x \in \mathbb{N}.(x = y)] \text{ IMPLIES } \exists z \in \mathbb{N}.(z = 0 \text{ OR } \forall w \in \mathbb{N}.\text{NOT } (w < 0))$$

$$[\exists x \in \mathbb{N}.(x = y)] \text{ IMPLIES } \exists z \in \mathbb{N}.\forall w \in \mathbb{N}.(z = 0 \text{ OR NOT } (w < 0))$$

$$\text{NOT } [\exists x \in \mathbb{N}.(x = y)] \text{ OR } \exists z \in \mathbb{N}.\forall w \in \mathbb{N}.(z = 0 \text{ OR NOT } (w < 0))$$

$$[\forall x \in \mathbb{N}.\text{NOT } (x = y)] \text{ OR } \exists z \in \mathbb{N}.\forall w \in \mathbb{N}.(z = 0 \text{ OR NOT } (w < 0))$$

Apply an alternative sequence of transformations

$$[\exists x \in \mathbb{N}.(x = y)] \text{ IMPLIES } \exists z \in \mathbb{N}.(z = 0 \text{ OR NOT } (\exists w \in \mathbb{N}.(w < 0)))$$

$$[\exists x \in \mathbb{N}.(x = y)] \text{ IMPLIES } \exists z \in \mathbb{N}.(z = 0 \text{ OR } \forall w \in \mathbb{N}.\text{NOT } (w < 0))$$

$$[\exists x \in \mathbb{N}.(x = y)] \text{ IMPLIES } \exists z \in \mathbb{N}.\forall w \in \mathbb{N}.(z = 0 \text{ OR NOT } (w < 0))$$

$$\text{NOT } [\exists x \in \mathbb{N}.(x = y)] \text{ OR } \exists z \in \mathbb{N}.\forall w \in \mathbb{N}.(z = 0 \text{ OR NOT } (w < 0))$$

$$[\forall x \in \mathbb{N}.\text{NOT } (x = y)] \text{ OR } \exists z \in \mathbb{N}.\forall w \in \mathbb{N}.(z = 0 \text{ OR NOT } (w < 0))$$

$$\exists z \in \mathbb{N}.([\forall x \in \mathbb{N}.\text{NOT } (x = y)] \text{ OR } \forall w \in \mathbb{N}.(z = 0 \text{ OR NOT } (w < 0)))$$

Apply an alternative sequence of transformations

$$[\exists x \in \mathbb{N}.(x = y)] \text{ IMPLIES } \exists z \in \mathbb{N}.(z = 0 \text{ OR NOT } (\exists w \in \mathbb{N}.(w < 0)))$$

$$[\exists x \in \mathbb{N}.(x = y)] \text{ IMPLIES } \exists z \in \mathbb{N}.(z = 0 \text{ OR } \forall w \in \mathbb{N}.\text{NOT } (w < 0))$$

$$[\exists x \in \mathbb{N}.(x = y)] \text{ IMPLIES } \exists z \in \mathbb{N}.\forall w \in \mathbb{N}.(z = 0 \text{ OR NOT } (w < 0))$$

$$\text{NOT } [\exists x \in \mathbb{N}.(x = y)] \text{ OR } \exists z \in \mathbb{N}.\forall w \in \mathbb{N}.(z = 0 \text{ OR NOT } (w < 0))$$

$$[\forall x \in \mathbb{N}.\text{NOT } (x = y)] \text{ OR } \exists z \in \mathbb{N}.\forall w \in \mathbb{N}.(z = 0 \text{ OR NOT } (w < 0))$$

$$\exists z \in \mathbb{N}.([\forall x \in \mathbb{N}.\text{NOT } (x = y)] \text{ OR } \forall w \in \mathbb{N}.(z = 0 \text{ OR NOT } (w < 0)))$$

$$\exists z \in \mathbb{N}.\forall w \in \mathbb{N}.([\forall x \in \mathbb{N}.\text{NOT } (x = y)] \text{ OR } (z = 0 \text{ OR NOT } (w < 0)))$$

Apply an alternative sequence of transformations

$[\exists x \in \mathbb{N}.(x = y)]$ IMPLIES $\exists z \in \mathbb{N}.(z = 0$ OR NOT $(\exists w \in \mathbb{N}.(w < 0)))$

$[\exists x \in \mathbb{N}.(x = y)]$ IMPLIES $\exists z \in \mathbb{N}.(z = 0$ OR $\forall w \in \mathbb{N}.$NOT $(w < 0))$

$[\exists x \in \mathbb{N}.(x = y)]$ IMPLIES $\exists z \in \mathbb{N}.\forall w \in \mathbb{N}.(z = 0$ OR NOT $(w < 0))$

NOT $[\exists x \in \mathbb{N}.(x = y)]$ OR $\exists z \in \mathbb{N}.\forall w \in \mathbb{N}.(z = 0$ OR NOT $(w < 0))$

$[\forall x \in \mathbb{N}.$NOT $(x = y)]$ OR $\exists z \in \mathbb{N}.\forall w \in \mathbb{N}.(z = 0$ OR NOT $(w < 0))$

$\exists z \in \mathbb{N}.([\forall x \in \mathbb{N}.$NOT $(x = y)]$ OR $\forall w \in \mathbb{N}.(z = 0$ OR NOT $(w < 0)))$

$\exists z \in \mathbb{N}.\forall w \in \mathbb{N}.([\forall x \in \mathbb{N}.$NOT $(x = y)]$ OR $(z = 0$ OR NOT $(w < 0)))$

$\exists z \in \mathbb{N}.\forall w \in \mathbb{N}.\forall x \in \mathbb{N}.[$NOT $(x = y)$ OR $(z = 0$ OR NOT $(w < 0))]$

CSC 240

Tutorial 2

January 20, 2021

True/False?

Recall from online lectures: "A PROPOSITIONAL FORMULA is an expression built up from Boolean variables using connectives such as

$$\mathrm{AND},\ \mathrm{OR},\ \mathrm{NOT},\ \mathrm{IMPLIES} \text{ and } \mathrm{XOR}.$$

It does not contain predicates or quantifiers."

True/False?

Recall from online lectures: "A PROPOSITIONAL FORMULA is an expression built up from Boolean variables using connectives such as

$$\mathrm{AND}, \mathrm{OR}, \mathrm{NOT}, \mathrm{IMPLIES} \text{ and } \mathrm{XOR}.$$

It does not contain predicates or quantifiers."

Claim: Every propositional formula can be written using only the connectives: AND, OR, NOT (and variables).

True/False?

Recall from online lectures: "A PROPOSITIONAL FORMULA is an expression built up from Boolean variables using connectives such as

$$\text{AND, OR, NOT, IMPLIES and XOR.}$$

It does not contain predicates or quantifiers."

Claim: Every propositional formula can be written using only the connectives: AND, OR, NOT (and variables).

True/False?

Complete Set of Connectives

Claim: Every propositional formula can be written using only the connectives AND, OR, NOT (and variables, and the constants T and F).

Complete Set of Connectives

Claim: Every propositional formula can be written using only the connectives AND , OR , NOT (and variables, and the constants T and F).

The claim is *true!* DNF (and CNF) does exactly that.

Complete Set of Connectives

Claim: Every propositional formula can be written using only the connectives AND , OR , NOT (and variables, and the constants T and F).

The claim is *true!* DNF (and CNF) does exactly that.

Such a set of connectives is called COMPLETE.

Complete Set of Connectives

Claim: Every propositional formula can be written using only the connectives AND, OR, NOT (and variables, and the constants T and F).

The claim is *true!* DNF (and CNF) does exactly that.

Such a set of connectives is called COMPLETE.

Formally: A set of connectives is COMPLETE if every propositional formula is logically equivalent to a propositional formula that only uses connectives from this set.

True/False?

Is $\{\mathrm{AND}, \mathrm{NOT}\}$ a complete set of connectives?

True/False?

Is $\{\text{AND}, \text{NOT}\}$ a complete set of connectives?

Yes!, because $\{\text{AND}, \text{OR}, \text{NOT}\}$ is a complete set of connectives, and

$$P \text{ OR } Q \equiv \text{NOT}(\text{NOT}(P) \text{ AND } \text{NOT}(Q)).$$

True/False?

Is $\{\text{AND}, \text{NOT}\}$ a complete set of connectives?

Yes!, because $\{\text{AND}, \text{OR}, \text{NOT}\}$ is a complete set of connectives, and

$$P \text{ OR } Q \equiv \text{NOT}(\text{NOT}(P) \text{ AND } \text{NOT}(Q)).$$

Is $\{\text{OR}, \text{NOT}\}$ a complete set of connectives?

True/False?

Is $\{\mathrm{AND}, \mathrm{NOT}\}$ a complete set of connectives?

Yes!, because $\{\mathrm{AND}, \mathrm{OR}, \mathrm{NOT}\}$ is a complete set of connectives, and

$$P \text{ OR } Q \equiv \mathrm{NOT}(\mathrm{NOT}(P) \text{ AND } \mathrm{NOT}(Q)).$$

Is $\{\mathrm{OR}, \mathrm{NOT}\}$ a complete set of connectives?

Of course; dual argument!

Can we do better?

Is $\{\mathrm{NOT}\}$ a complete set of connectives?

Can we do better?

Is $\{\mathrm{NOT}\}$ a complete set of connectives?

Can't express formulas with more than one propositional variable: for example, x AND y.

Can we do better?

Is {NOT} a complete set of connectives?

Can't express formulas with more than one propositional variable: for example, x AND y.

Is {AND} a complete set of connectives?

Can we do better?

Is $\{\mathrm{NOT}\}$ a complete set of connectives?

Can't express formulas with more than one propositional variable: for example, x AND y.

Is $\{\mathrm{AND}\}$ a complete set of connectives?

Can't express $\mathrm{NOT}(x)$. Why?

Can we do better?

Is $\{\mathrm{NOT}\}$ a complete set of connectives?

Can't express formulas with more than one propositional variable: for example, $x \ \mathrm{AND} \ y$.

Is $\{\mathrm{AND}\}$ a complete set of connectives?

Can't express $\mathrm{NOT}(x)$. Why?
Suppose P is a propositional formula involving only propositional variables and the connective AND. Consider the truth assignment making all propositional variable T. Then $P = T$, but $\mathrm{NOT}(x) = F$.

AND , OR

So we've seen $\{\mathrm{NOT}, \mathrm{AND}\}$ and $\{\mathrm{NOT}, \mathrm{OR}\}$ are each a complete set of connectives; while $\{\mathrm{NOT}\}$, $\{\mathrm{AND}\}$, $\{\mathrm{OR}\}$ are not.

AND , OR

So we've seen $\{\text{NOT}, \text{AND}\}$ and $\{\text{NOT}, \text{OR}\}$ are each a complete set of connectives; while $\{\text{NOT}\}$, $\{\text{AND}\}$, $\{\text{OR}\}$ are not.

What about $\{\text{AND}, \text{OR}\}$?

AND , OR

So we've seen $\{\text{NOT}, \text{AND}\}$ and $\{\text{NOT}, \text{OR}\}$ are each a complete set of connectives; while $\{\text{NOT}\}$, $\{\text{AND}\}$, $\{\text{OR}\}$ are not.

What about $\{\text{AND}, \text{OR}\}$?

NO!

AND , OR

So we've seen $\{\text{NOT}, \text{AND}\}$ and $\{\text{NOT}, \text{OR}\}$ are each a complete set of connectives; while $\{\text{NOT}\}$, $\{\text{AND}\}$, $\{\text{OR}\}$ are not.

What about $\{\text{AND}, \text{OR}\}$?

NO! $\text{NOT}(x)$ cannot be expressed. Why?

AND , OR

So we've seen {NOT, AND} and {NOT, OR} are each a complete set of connectives; while {NOT}, {AND}, {OR} are not.

What about {AND, OR}?

NO! $\mathrm{NOT}(x)$ cannot be expressed. Why?Suppose P is a propositional formula involving only the propositional variables and the connectives AND and OR. Consider the truth assignment making all propositional variable T. Then $P = T$, but $\mathrm{NOT}(x) = F$.

Can we do better?

Is $\{\mathrm{NAND}\}$ a complete set of connectives?

NAND is defined by the following truth table:

P	Q	P NAND Q
T	T	F
T	F	T
F	T	T
F	F	T

Can we do better?

Is $\{\text{NAND}\}$ a complete set of connectives?

NAND is defined by the following truth table:

P	Q	P NAND Q
T	T	F
T	F	T
F	T	T
F	F	T

Yes! Because $\{\text{NOT}, \text{AND}\}$ is a complete set of connectives and

$$P \text{ NAND } P = \text{NOT}(P)$$

$$\begin{aligned}(P \text{ NAND } Q) \text{ NAND } (P \text{ NAND } Q) &\equiv \text{NOT}(P \text{ NAND } Q) \\ &\equiv P \text{ AND } Q.\end{aligned}$$

Ternary Connective

We introduce a new ternary connective: IF-THEN-ELSE(x, y, z) = "IF x THEN y ELSE z".

Ternary Connective

We introduce a new ternary connective: IF-THEN-ELSE(x, y, z) = "IF x THEN y ELSE z".

You should be able to both: build a truth table directly from this informal description; and translate this informal description to a well-formed logical formula using only our familiar binary connectives.

Ternary Connective

We introduce a new ternary connective: IF-THEN-ELSE(x, y, z) = "IF x THEN y ELSE z".

You should be able to both: build a truth table directly from this informal description; and translate this informal description to a well-formed logical formula using only our familiar binary connectives.

{IF-THEN-ELSE} is not a complete set of connectives. But {IF-THEN-ELSE, T, F} is. We regard T and F is 0-ary connectives; i.e., constants.

IF-THEN-ELSE

$$\mathrm{NOT}(x) \equiv \text{IF-THEN-ELSE}(x, F, T)$$

IF-THEN-ELSE

$$\mathrm{NOT}(x) \equiv \text{IF-THEN-ELSE}(x, F, T)$$

$$\mathrm{AND}(x, y) \equiv \text{IF-THEN-ELSE}(x, y, x)$$

IF-THEN-ELSE

$$\mathrm{NOT}(x) \equiv \textsf{IF-THEN-ELSE}(x, F, T)$$

$$\mathrm{AND}(x, y) \equiv \textsf{IF-THEN-ELSE}(x, y, x)$$

$$\mathrm{OR}(x, y) \equiv \textsf{IF-THEN-ELSE}(x, x, y)$$

Challenges

In ascending order of difficulty:

- Is $\{\mathrm{NOT}, \mathrm{IMPLIES}\}$ complete?

Challenges

In ascending order of difficulty:

- Is $\{\text{NOT}, \text{ IMPLIES}\}$ complete?
- Is $\{F, \text{ IMPLIES}\}$ complete?

Challenges

In ascending order of difficulty:

- Is $\{\text{NOT}, \text{IMPLIES}\}$ complete?
- Is $\{F, \text{IMPLIES}\}$ complete?
- Is $\{\text{AND}, \text{IMPLIES}\}$ complete?

Challenges

In ascending order of difficulty:

- Is $\{\text{NOT}, \text{ IMPLIES}\}$ complete?
- Is $\{F, \text{ IMPLIES}\}$ complete?
- Is $\{\text{AND}\ ,\ \text{IMPLIES}\}$ complete?
- Let MINORITY be a ternary connective disagreeing with the majority of truth-values (you should formalize this with a truth table!). Is $\{\text{MINORITY}, F\}$ complete?

Challenges

In ascending order of difficulty:

- Is $\{\mathrm{NOT},\ \mathrm{IMPLIES}\}$ complete?
- Is $\{F,\ \mathrm{IMPLIES}\}$ complete?
- Is $\{\mathrm{AND}\ ,\ \mathrm{IMPLIES}\}$ complete?
- Let MINORITY be a ternary connective disagreeing with the majority of truth-values (you should formalize this with a truth table!). Is $\{\text{MINORITY}, F\}$ complete?
- Is $\{\text{XOR}\}$ complete? (Hint: parity!)

Challenges

In ascending order of difficulty:

- Is $\{\mathrm{NOT},\ \mathrm{IMPLIES}\}$ complete?
- Is $\{F,\ \mathrm{IMPLIES}\}$ complete?
- Is $\{\mathrm{AND}\ ,\ \mathrm{IMPLIES}\}$ complete?
- Let MINORITY be a ternary connective disagreeing with the majority of truth-values (you should formalize this with a truth table!). Is $\{\text{MINORITY}, F\}$ complete?
- Is $\{\text{XOR}\}$ complete? (Hint: parity!)
- Is $\{\text{MINORITY}\}$ complete?

Challenges

In ascending order of difficulty:

- Is $\{\text{NOT}, \text{ IMPLIES}\}$ complete?
- Is $\{F, \text{ IMPLIES}\}$ complete?
- Is $\{\text{AND} \text{ , } \text{IMPLIES}\}$ complete?

- Let MINORITY be a ternary connective disagreeing with the majority of truth-values (you should formalize this with a truth table!). Is $\{\text{MINORITY}, F\}$ complete?
- Is $\{\text{XOR}\}$ complete? (Hint: parity!)

- Is $\{\text{MINORITY}\}$ complete?
- Show that $\{\text{AND}, \text{IFF}, \text{XOR}\}$ is a minimal complete set of connectives (i.e., it is complete and no proper subset of it is complete).

Solutions to

CSC240 Winter 2021 Homework Assignment 2

1. (a) Consider the truth table of any propositional formula $G \in \mathcal{A}$. There are 8 different rows in the truth table. Each row for which $G = \text{T}$ contributes a conjunction of 3 literals, which are all different, to the DNF of G. Hence, the DNF for G using the method described in Online Lecture 3 contains at most 24 occurrences of the variables X, Y, and Z.

   To reduce the number of occurrences of variables, group the rows into pairs that agree on the values of X and Y, but disagree on the value of Z. If $G = \text{F}$ on both these rows, then they contribute nothing to the DNF. If $G = \text{T}$ on exactly one of these rows, then that row contributes a total of three occurrences of variables. If $G = \text{T}$ on both these rows, then together, they can contribute a conjunction of 2 literals, instead of two conjunctions of 3 literals each. For example, consider the two rows in which X and Y are both T. Then
   (X AND Y AND Z) OR (X AND Y AND NOT Z)
   is logically equivalent to
   (X AND Y) AND (Z OR NOT Z), by distributivity,
   which is logically equivalent to
   (X AND Y) AND T,
   which is logically equivalent to
   (X AND Y).

   Since there are 4 pairs of rows in the truth table and each pair of rows either contributes no conjunction or contributes a conjunction with at most 3 literals, there is a logically equivalent formula $P \in \mathcal{A}$ with $sz(P) \leq 12$.

   (b) Consider the propositional formula X XOR Y XOR Z. Using the method from Online Lecture 3, a logically equivalent propositional formula in disjunctive normal form is

$$\begin{aligned} Q \quad = \quad & \text{“}(X \text{ AND } Y \text{ AND } Z) \text{ OR } (X \text{ AND NOT}(Y) \text{ AND NOT}(Z)) \text{ OR} \\ & (\text{NOT}(X) \text{ AND } Y \text{ AND NOT}(Z)) \text{ OR } (\text{NOT}(X) \text{ AND NOT}(Y) \text{ AND } Z)\text{”}. \end{aligned}$$

   Note that $sz(Q) = 12$. Q is true when an odd number of the variables X, Y, and Z are true. This means that any conjunction of literals in a logically equivalent DNF formula must contain all 3 different variables. Otherwise, set the variables that occur in the conjunction so that it is true. Changing the truth value of the missing variable does not change the value of the conjunction to false. But then the DNF formula is true in at least one case when an even number of the variables are true. This means it is not logically equivalent to X XOR Y XOR Z.

   Since there are four different truth assignments in which an odd number of variables are true and each conjunction of three literals is true for only one truth assignment, there must be four conjunctions of three literals in any logically equivalent DNF formula P. Hence $sz(P) \geq 12$.

   (c) If $G \in \mathcal{A}$, then so is NOT G. By part (a), there is a propositional formula $P \in \mathcal{A}$ in disjunctive normal form that is logically equivalent to NOT G such that $sz(P) \leq$

12. Applying DeMorgan's Laws to NOT P gives a propositional formula $Q \in \mathcal{A}$ in conjunctive normal form that is logically equivalent to G and such that $sz(Q) = sz(P)$.

2. (a) Let $D = \mathbb{N}$,
$p(x) =$ "$x \geq 0$", and
$q(x) =$ "NOT $p(x)$".

Every natural number x is at least 0, so $p(x)$ is true and $q(x)$ is false. Hence $p(x)$ IMPLIES $q(x)$ is false, so $\exists x \in D.(p(x)$ IMPLIES $q(x))$ is false. Thus, NOT $\exists x \in D.(p(x)$ IMPLIES $q(x))$ is true, i.e. R is true.

For every natural number x, NOT $q(x)$ is true, so $p(x)$ IMPLIES NOT $q(x)$ is true. Thus $\exists x \in D.(p(x)$ IMPLIES NOT $q(x))$ is true, i.e. S is true.

(b) This is impossible. Consider any interpretation in which S is false. Then, for every $x \in D$, $p(x)$ IMPLIES NOT $q(x)$ is false, so $p(x)$ is true and NOT $q(x)$ is false.

Since D is nonempty, there is some element $d \in D$. Then $p(d)$ is true and NOT $q(d)$ is false, so $q(d)$ is true. Thus $p(d)$ IMPLIES $q(d)$ is true. This implies that $\exists x \in D.(p(x)$ IMPLIES $q(x))$ is true. Hence NOT $\exists x \in D.(p(x)$ IMPLIES $q(x))$ is false, i.e. R is false.

(c) Let $D = \mathbb{N}$,
$p(x) =$ "x is even", and
$q(x) =$ "x is divisible by 4".

Consider $x = 1$. Since $p(1)$ is false, $p(1)$ IMPLIES $q(1)$ is true. Hence, $\exists x \in D.(p(x)$ IMPLIES $q(x))$ is true and NOT $\exists x \in D.(p(x)$ IMPLIES $q(x))$ is false, i.e. R is false.

Consider $x = 2$. Since $q(2)$ is false, NOT $q(2)$ is true, so $p(2)$ IMPLIES NOT $q(2)$ is true. Thus $\exists x \in D.(p(x)$ IMPLIES $q(x))$ is true, i.e. S is true.

(d) Let $D = \{\text{T}, \text{F}\}$,
$p(x) =$ "T", and
$q(x) =$ "T".

Hence, for all $x \in D$, $p(x)$ IMPLIES $q(x)$ is true and $p(x)$ IMPLIES NOT $q(x)$ is false. Thus, $\exists x \in D.(p(x)$ IMPLIES $q(x))$ is true, which implies that NOT $\exists x \in D.(p(x)$ IMPLIES $q(x))$ is false, i.e. R is false. Furthermore, $\exists x \in D.(p(x)$ IMPLIES NOT $q(x))$ is false, i.e. S is false.

Lecture 5

A proposition is a statement that is either true or false.

An axiom is a proposition that we agree is true.

A proof is a convincing argument that a proposition is true.
It consists of a sequence of axioms, previously proved propositions, and logical deductions.

A logical deduction uses an inference rule to prove a new proposition from axioms and previously proved propositions.

Substitution

Let R be a tautology that contains propositional variable P.
If R' is the formula obtained by replacing
every occurrence of P in R by the formula (Q),
then R' is a tautology.

Example:
$R =$ (A OR P) IMPLIES (P OR A)
$Q =$ C AND D
$R' =$ (A OR (C AND D)) IMPLIES ((C AND D) OR A)

Substitution

Let R be a tautology that contains propositional variable P.
If R' is the formula obtained by replacing
every occurrence of P in R by the formula (Q),
then R' is a tautology.

Example:
$R =$ (A OR P) IMPLIES (P OR A)
$Q =$ C AND D
(A OR (C AND D)) IMPLIES (P OR A) is not a tautology

Substitution

Let R be a tautology that contains propositional variable P.
If R' is the formula obtained by replacing
every occurrence of P in R by the formula (Q),
then R' is a tautology.

Example:
$R =$ (A OR P) IMPLIES (P OR A)
$Q = \forall e \in E.a(e)$
$R' =$ (A OR $(\forall e \in E.a(e))$) IMPLIES ($(\forall e \in E.a(e))$ OR A)

Substitution

Let S' be a formula that is logically equivalent to S.
If S is a subformula of R and
R' is a formula obtained by replacing
some occurrences of S in R by S'
then R' is logically equivalent to R.

Example:
$S =$ NOT(A) AND NOT(B)
$S' =$ NOT(A OR B)
$R =$
(NOT(A) AND NOT(B)) XOR (B IFF (NOT(A) AND NOT(B))

Substitution

Let S' be a formula that is logically equivalent to S.
If S is a subformula of R and
R' is a formula obtained by replacing
some occurrences of S in R by S',
then R' is logically equivalent to R.

Example:
$S =$ NOT(A) AND NOT(B)
$S' =$ NOT(A OR B)
$R =$
(NOT(A) AND NOT(B)) XOR (B IFF (NOT(A) AND NOT(B))
is logically equivalent to
$R' =$ NOT(A OR B) XOR (B IFF (NOT(A) AND NOT(B))

Substitution

Let S' be a formula that is logically equivalent to S.
If S is a subformula of R and
R' is a formula obtained by replacing
some occurrences of S in R by S',
then R' is logically equivalent to R.

Example:
$S =$ NOT(A) AND NOT(B)
$S' =$ NOT(A OR B)
$R =$
(NOT(A) AND NOT(B)) XOR (B IFF (NOT(A) AND NOT(B)))
is logically equivalent to
$R' =$ (NOT(A) AND NOT(B)) XOR (B IFF (NOT(A OR B)))

Substitution

Let S' be a formula that is logically equivalent to S.
If S is a subformula of R and
R' is a formula obtained by replacing
some occurrences of S in R by S',
then R' is logically equivalent to R.

Example:
$S =$ NOT(A) AND NOT(B)
$S' =$ NOT(A OR B)
$R =$
(NOT(A) AND NOT(B)) XOR (B IFF (NOT(A) AND NOT(B))
is logically equivalent to
$R' =$ (NOT(A OR B)) XOR (B IFF (NOT(A OR B))

Modus Ponens

If P and P IMPLIES Q are axioms or previously proved propositions, then Q is true.

Example:

1. $\forall e \in E.\ell(e)$ axiom
2. $(\forall e \in E.\ell(e))$ IMPLIES $\exists e \in E.\ell(e)$ tautology
3. $\exists e \in E.\ell(e)$ modus ponens 1,2

Modus Ponens

If P and P IMPLIES Q are axioms or previously proved propositions, then Q is true.

Example:

1. $\forall e \in E.\ell(e)$ axiom
2. $(\forall e \in E.\ell(e))$ IMPLIES $\exists e \in E.\ell(e)$ tautology
3. $\exists e \in E.\ell(e)$ modus ponens 1,2

Modus Ponens

If P and P IMPLIES Q are axioms or previously proved propositions, then Q is true.

Example:

1. $\forall e \in E.\ell(e)$ axiom
2. $(\forall e \in E.\ell(e))$ IMPLIES $\exists e \in E.\ell(e)$ tautology
3. $\exists e \in E.\ell(e)$ modus ponens 1,2

Modus Ponens

If P and P IMPLIES Q are axioms or previously proved propositions, then Q is true.

Example:

1. $\forall e \in E.\ell(e)$ axiom
2. $(\forall e \in E.\ell(e))$ IMPLIES $\exists e \in E.\ell(e)$ tautology
3. $\exists e \in E.\ell(e)$ modus ponens 1,2

Modus Ponens

If P and P IMPLIES Q are axioms or previously proved propositions, then Q is true.

Example:

1. $\forall e \in E.\ell(e)$ axiom
2. $(\forall e \in E.\ell(e))$ IMPLIES $\exists e \in E.\ell(e)$ tautology
3. $\exists e \in E.\ell(e)$ modus ponens 1,2

Formal Proof:

- Number each line.
- Write one proposition per line.
- Justify each line.

Modus Ponens

If P and P IMPLIES Q are true propositions,
then Q is a true proposition.

Not an example:
If he is a criminal, he has something to hide.
He has something to hide.
Therefore he is a criminal.

From the axioms P IMPLIES Q and Q, you can't conclude P.

Specialization

If $c \in D$ and
$\forall x \in D.a(x)$ is true,
then $a(c)$ is true.

Specialization

If $c \in D$ and
$\forall x \in D.a(x)$ is true,
then $a(c)$ is true.

If $c \in D$, then $(\forall x \in D.a(x))$ IMPLIES $a(c)$ is a tautology.
If $\forall x \in D.a(x)$ is an axiom,
then $a(c)$ follows by modus ponens.

Transitivity

If P IMPLIES Q and Q IMPLIES R are axioms or previously proved propositions, then P IMPLIES R is true.

Example:
If you study hard, you'll learn the material.
If you learn the material, you'll pass the course.
Therefore, if you study hard, you'll pass the course.

Direct Proof of Implication

Assume P

⋮

Q

Therefore P IMPLIES Q.

Direct Proof of Implication

Assume P
⋮
Q

Therefore P IMPLIES Q.

Example: Proof of Transitivity

1. Assume P
2. P IMPLIES Q axiom
3. Q modus ponens 1,2
4. Q IMPLIES R axiom
5. R modus ponens 3,4
6. Therefore P IMPLIES R direct proof 1,5

Direct Proof of Implication

Assume P
⋮
Q
Therefore P IMPLIES Q.

Example: Proof of Transitivity

1. Assume P
2. P IMPLIES Q axiom
3. Q modus ponens 1,2
4. Q IMPLIES R axiom
5. R modus ponens 3,4
6. Therefore P IMPLIES R direct proof 1,5

Direct Proof of Implication

Assume P

$\vdots$

Q

Therefore P IMPLIES Q.

Example: Proof of Transitivity

1. Assume P
2. P IMPLIES Q axiom
3. Q modus ponens 1,2
4. Q IMPLIES R axiom
5. R modus ponens 3,4
6. Therefore P IMPLIES R direct proof 1,5

Direct Proof of Implication

Assume P

⋮

Q

Therefore P IMPLIES Q.

Example: Proof of Transitivity

1. Assume P
2. P IMPLIES Q axiom
3. Q modus ponens 1,2
4. Q IMPLIES R axiom
5. R modus ponens 3,4

6. Therefore P IMPLIES R direct proof 1,5

Direct Proof of Implication

Assume P
⋮
Q

Therefore P IMPLIES Q.

Example: Proof of Transitivity

1. Assume P
2. P IMPLIES Q axiom
3. Q modus ponens 1,2
4. Q IMPLIES R axiom
5. R modus ponens 3,4
6. Therefore P IMPLIES R direct proof 1,5

Direct Proof of Implication

Assume P

⋮

Q

Therefore P IMPLIES Q.

Example: Proof of Transitivity

1. Assume P
2. P IMPLIES Q axiom
3. Q modus ponens 1,2
4. Q IMPLIES R axiom
5. R modus ponens 3,4
6. Therefore P IMPLIES R direct proof 1,5

Direct Proof of Implication

Assume P
$\vdots$
Q
Therefore P IMPLIES Q.

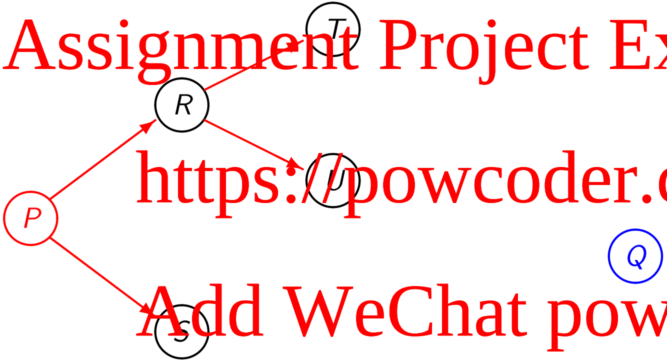
Example: Proof of Transitivity

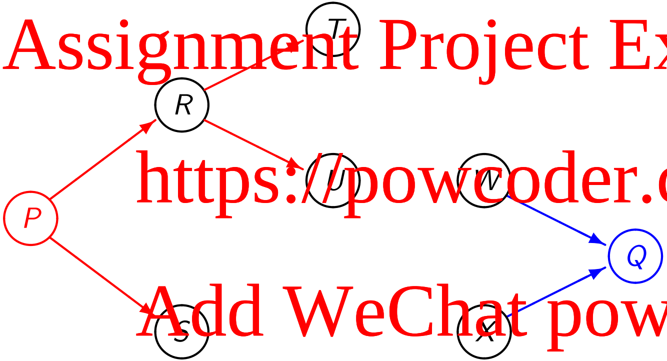
1. Assume P
2. P IMPLIES Q axiom
3. Q modus ponens 1,2
4. Q IMPLIES R axiom
5. R modus ponens 3,4
6. Therefore P IMPLIES R direct proof 1,5

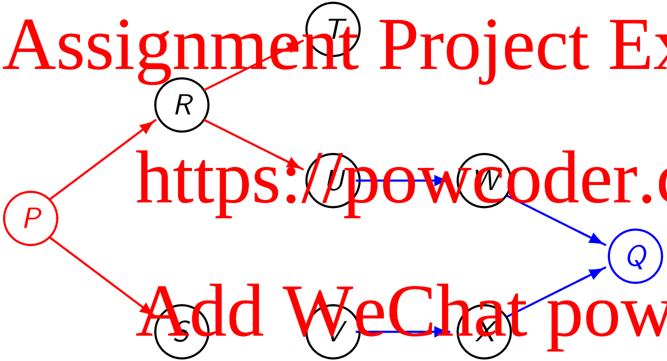
P

Q


R
P
Q
S


P
R
S
T
U
Q


T
R
U
W
P
Q
S
X


T
R
U
W
P
Q
S
V
X

Indirect Proof of Implication

P IMPLIES Q is logically equivalent to NOT(Q) IMPLIES NOT(P), so proving NOT(Q) IMPLIES NOT(P) proves P IMPLIES Q.

Assume NOT(Q)

⋮

NOT(P)

Hence NOT(Q) IMPLIES NOT(P)

Therefore P IMPLIES Q.

LEMMA: If x is even, then x^2 is even.

Proof:
Assume x is even.
Then $x = 2k$ for some integer k
so $x^2 = (2k)^2 = 2 \times (2k^2)$, which is even. ■

LEMMA: If x is even, then x^2 is even.

Proof:

Assume x is even.
Then $x = 2k$ for some integer k
so $x^2 = (2k)^2 = 2 \times (2k^2)$, which is even. ■

LEMMA: If x is even, then x^2 is even.

Proof:

Assume x is even.

Then $x = 2k$ for some integer k

so $x^2 = (2k)^2 = 2 \times (2k^2)$, which is even. ■

LEMMA: If x is even, then x^2 is even.

Proof:

Assume x is even.

Then $x = 2k$ for some integer k

so $x^2 = (2k)^2 = 2 \times (2k^2)$, which is even. ■

LEMMA: If x is even, then x^2 is even.

Proof:
Assume x is even.
Then $x = 2k$ for some integer k,
so $x^2 = (2k)^2 = 2 \times (2k^2)$, which is even. ■

LEMMA: If x^2 is even, then x is even.

Proof:
Suppose x^2 is even.
Then $x^2 = 2k$ for some integer k.
?

LEMMA: If x^2 is even, then x is even.

Proof:

Suppose x is not even.

Then x is odd,

so $x = 2k + 1$ for some integer k.

Hence $x^2 = (2k + 1)^2 = 4k^2 + 4k + 1 = 2(2k^2 + 2k) + 1$,

which is odd.

Therefore x^2 is not even. ■

LEMMA: If x^2 is even, then x is even.

Proof:

Suppose x is not even.

Then x is odd,

so $x = 2k + 1$ for some integer k.

Hence $x^2 = (2k+1)^2 = 4k^2 + 4k + 1 = 2(2k^2 + 2k) + 1$,

which is odd.

Therefore x^2 is not even. ■

LEMMA: If x^2 is even, then x is even.

Proof:

Suppose x is not even.

Then x is odd,

so $x = 2k + 1$ for some integer k.

Hence $x^2 = (2k + 1)^2 = 4k^2 + 4k + 1 = 2(2k^2 + 2k) + 1$,

which is odd.

Therefore x^2 is not even. ■

LEMMA: If x^2 is even, then x is even.

Proof:

Suppose x is not even.

Then x is odd,

so $x = 2k + 1$ for some integer k.

Hence $x^2 = (2k+1)^2 = 4k^2 + 4k + 1 = 2(2k^2 + 2k) + 1$,

which is odd.

Therefore x^2 is not even. ■

LEMMA: If x^2 is even, then x is even.

Proof:

Suppose x is not even.

Then x is odd,

so $x = 2k + 1$ for some integer k.

Hence $x^2 = (2k+1)^2 = 4k^2 + 4k + 1 = 2(2k^2 + 2k) + 1$,

which is odd.

Therefore x^2 is not even. ■

LEMMA: If x^2 is even, then x is even.

Proof:

Suppose x is not even.

Then x is odd,

so $x = 2k + 1$ for some integer k.

Hence $x^2 = (2k+1)^2 = 4k^2 + 4k + 1 = 2(2k^2 + 2k) + 1$,

which is odd.

Therefore x^2 is not even. ■

LEMMA: If x^2 is even, then x is even.

Proof:

Suppose x is not even.

Then x is odd,

so $x = 2k + 1$ for some integer k.

Hence $x^2 = (2k+1)^2 = 4k^2 + 4k + 1 = 2(2k^2 + 2k) + 1$,

which is odd.

Therefore x^2 is not even. ■

Proof of a Disjunction

⋮

P

Therefore P OR Q.

Proof of a Conjunction

$\vdots$
P
$\vdots$
Q
Therefore P AND Q.

Use of Conjunction

P AND Q
P

Use of Conjunction

P AND Q
(P AND Q) IMPLIES P tautology
P

Proof by Contradiction

Q AND NOT(Q) is a contradiction,
so NOT(Q AND NOT(Q)) is a tautology.

Assume NOT(P)

⋮

Q

⋮

NOT(Q)

Q AND NOT(Q) proof of conjunction

NOT(Q AND NOT(Q)) IMPLIES P indirect proof

NOT(Q AND NOT(Q)) tautology

Therefore P modus ponens

THEOREM: $\sqrt{2}$ is irrational.

Proof:
To obtain a contradiction, assume $\sqrt{2}$ is rational.
Then there exist relatively prime positive integers x and y,
such that $\sqrt{2} = x/y$.
Since $x^2 = (\sqrt{2}y)^2 = 2y^2$, x^2 is even.
From the lemma, x is even.
Thus, there exists $k \in \mathbb{Z}^+$ such that $x = 2k$,
so $2y^2 = x^2 = (2k)^2 = 4k^2$ and $y^2 = 2k^2$.
Therefore y^2 is even.
From the lemma, y is even.
Since 2 divides both x and y, they are not relatively prime.
This is a contradiction. Hence $\sqrt{2}$ is irrational. ■

THEOREM: $\sqrt{2}$ is irrational.

Proof:
To obtain a contradiction, assume $\sqrt{2}$ is rational.
Then there exist relatively prime positive integers x and y,
such that $\sqrt{2} = x/y$.
Since $x^2 = (\sqrt{2}y)^2 = 2y^2$, x^2 is even.
From the lemma, x is even.
Thus, there exists $k \in \mathbb{Z}^+$ such that $x = 2k$,
so $2y^2 = x^2 = (2k)^2 = 4k^2$ and $y^2 = 2k^2$.
Therefore y^2 is even.
From the lemma, y is even.
Since 2 divides both x and y, they are not relatively prime.
This is a contradiction. Hence $\sqrt{2}$ is irrational. $\blacksquare$

THEOREM: $\sqrt{2}$ is irrational.

Proof:
To obtain a contradiction, assume $\sqrt{2}$ is rational.
Then there exist relatively prime positive integers x and y,
such that $\sqrt{2} = x/y$.
Since $x^2 = (\sqrt{2}y)^2 = 2y^2$, x^2 is even.
From the lemma, x is even.
Thus, there exists $k \in \mathbb{Z}^+$ such that $x = 2k$,
so $2y^2 = x^2 = (2k)^2 = 4k^2$ and $y^2 = 2k^2$.
Therefore y^2 is even.
From the lemma, y is even.
Since 2 divides both x and y, they are not relatively prime.
This is a contradiction. Hence $\sqrt{2}$ is irrational. ■

THEOREM: $\sqrt{2}$ is irrational.

Proof:
To obtain a contradiction, assume $\sqrt{2}$ is rational.
Then there exist relatively prime positive integers x and y,
such that $\sqrt{2} = x/y$.
Since $x^2 = (\sqrt{2}y)^2 = 2y^2$, x^2 is even.
From the lemma, x is even.
Thus, there exists $k \in \mathbb{Z}^+$ such that $x = 2k$,
so $2y^2 = x^2 = (2k)^2 = 4k^2$ and $y^2 = 2k^2$.
Therefore y^2 is even.
From the lemma, y is even.
Since 2 divides both x and y, they are not relatively prime.
This is a contradiction. Hence $\sqrt{2}$ is irrational. ■

THEOREM: $\sqrt{2}$ is irrational.

Proof:
To obtain a contradiction, assume $\sqrt{2}$ is rational.
Then there exist relatively prime positive integers x and y,
such that $\sqrt{2} = x/y$.
Since $x^2 = (\sqrt{2}y)^2 = 2y^2$, x^2 is even.
From the lemma, x is even.
Thus, there exists $k \in \mathbb{Z}^+$ such that $x = 2k$,
so $2y^2 = x^2 = (2k)^2 = 4k^2$ and $y^2 = 2k^2$.
Therefore y^2 is even.
From the lemma, y is even.
Since 2 divides both x and y, they are not relatively prime.
This is a contradiction. Hence $\sqrt{2}$ is irrational. $\blacksquare$

THEOREM: $\sqrt{2}$ is irrational.

Proof:
To obtain a contradiction, assume $\sqrt{2}$ is rational.
Then there exist relatively prime positive integers x and y,
such that $\sqrt{2} = x/y$.
Since $x^2 = (\sqrt{2}y)^2 = 2y^2$, x^2 is even.
From the lemma, x is even.
Thus, there exists $k \in \mathbb{Z}^+$ such that $x = 2k$,
so $2y^2 = x^2 = (2k)^2 = 4k^2$ and $y^2 = 2k^2$.
Therefore y^2 is even.
From the lemma, y is even.
Since 2 divides both x and y, they are not relatively prime.
This is a contradiction. Hence $\sqrt{2}$ is irrational. ■

THEOREM: $\sqrt{2}$ is irrational.

Proof:

To obtain a contradiction, assume $\sqrt{2}$ is rational.

Then there exist relatively prime positive integers x and y,

such that $\sqrt{2} = x/y$.

Since $x^2 = (\sqrt{2}y)^2 = 2y^2$, x^2 is even.

From the lemma, x is even.

Thus, there exists $k \in \mathbb{Z}^+$ such that $x = 2k$,

so $2y^2 = x^2 = (2k)^2 = 4k^2$ and $y^2 = 2k^2$.

Therefore y^2 is even.

From the lemma, y is even.

Since 2 divides both x and y, they are not relatively prime.

This is a contradiction. Hence $\sqrt{2}$ is irrational. ■

THEOREM: $\sqrt{2}$ is irrational.

Proof:
To obtain a contradiction, assume $\sqrt{2}$ is rational.
Then there exist relatively prime positive integers x and y,
such that $\sqrt{2} = x/y$.
Since $x^2 = (\sqrt{2}y)^2 = 2y^2$, x^2 is even.
From the lemma, x is even.
Thus, there exists $k \in \mathbb{Z}^+$ such that $x = 2k$,
so $2y^2 = x^2 = (2k)^2 = 4k^2$ and $y^2 = 2k^2$.
Therefore y^2 is even.
From the lemma, y is even.
Since 2 divides both x and y, they are not relatively prime.
This is a contradiction. Hence $\sqrt{2}$ is irrational. ∎

THEOREM: $\sqrt{2}$ is irrational.

Proof:
To obtain a contradiction, assume $\sqrt{2}$ is rational.
Then there exist relatively prime positive integers x and y,
such that $\sqrt{2} = x/y$.
Since $x^2 = (\sqrt{2}y)^2 = 2y^2$, x^2 is even.
From the lemma, x is even.
Thus, there exists $k \in \mathbb{Z}^+$ such that $x = 2k$,
so $2y^2 = x^2 = (2k)^2 = 4k^2$ and $y^2 = 2k^2$.
Therefore y^2 is even.
From the lemma, y is even.
Since 2 divides both x and y, they are not relatively prime.
This is a contradiction. Hence $\sqrt{2}$ is irrational. ■

THEOREM: $\sqrt{2}$ is irrational.

Proof:
To obtain a contradiction, assume $\sqrt{2}$ is rational.
Then there exist relatively prime positive integers x and y,
such that $\sqrt{2} = x/y$.
Since $x^2 = (\sqrt{2}y)^2 = 2y^2$, x^2 is even.
From the lemma, x is even.
Thus, there exists $k \in \mathbb{Z}^+$ such that $x = 2k$,
so $2y^2 = x^2 = (2k)^2 = 4k^2$ and $y^2 = 2k^2$.
Therefore y^2 is even.
From the lemma, y is even.
Since 2 divides both x and y, they are not relatively prime.
This is a contradiction. Hence $\sqrt{2}$ is irrational. ■

THEOREM: $\sqrt{2}$ is irrational.

Proof:
To obtain a contradiction, assume $\sqrt{2}$ is rational.
Then there exist relatively prime positive integers x and y,
such that $\sqrt{2} = x/y$.
Since $x^2 = (\sqrt{2}y)^2 = 2y^2$, x^2 is even.
From the lemma, x is even.
Thus, there exists $k \in \mathbb{Z}^+$ such that $x = 2k$,
so $2y^2 = x^2 = (2k)^2 = 4k^2$ and $y^2 = 2k^2$.
Therefore y^2 is even.
From the lemma, y is even.
Since 2 divides both x and y, they are not relatively prime.
This is a contradiction. Hence $\sqrt{2}$ is irrational. ∎

Proof of Equivalences

P IFF Q is logically equivalent to
(P IMPLIES Q) AND (Q IMPLIES P)
Prove P IMPLIES Q and Q IMPLIES P separately.

To prove P IFF Q, Q IFF R, and P IFF R
it suffices to prove
P IMPLIES Q, Q IMPLIES R, and R IMPLIES P.

Proving P IMPLIES Q, Q IMPLIES R, and P IMPLIES R
is not sufficent.

$\vdots$

P_1 IMPLIES Q

$\vdots$

P_n IMPLIES Q

Therefore $(\mathrm{P}_1$ OR $\cdots$ OR $\mathrm{P}_n)$ IMPLIES Q

The cases $\mathrm{P}_1, \ldots, \mathrm{P}_n$ can overlap.

Proof by Cases

P_1 OR $\cdots$ OR P_n

$\vdots$

P_1 IMPLIES Q

$\vdots$

P_n IMPLIES Q

Therefore (P_1 OR $\cdots$ OR P_n) IMPLIES Q

Hence Q.

Proof by Cases

P IMPLIES $(\mathrm{P}_1$ OR $\cdots$ OR $\mathrm{P}_n)$

$\vdots$

P_1 IMPLIES Q

$\vdots$

P_n IMPLIES Q

Therefore $(\mathrm{P}_1$ OR $\cdots$ OR $\mathrm{P}_n)$ IMPLIES Q.

Hence P IMPLIES Q.

LEMMA: P IMPLIES ((P AND C) OR (P AND NOT(C)))

1. Assume P
2. C OR NOT C tautology
    3. Assume C
    4. P AND C proof of conjunction 1,3
    5. (P AND C) OR (P AND NOT(C)) proof of disjunction 4
6. C IMPLIES ((P AND C) OR (P AND NOT(C))) direct proof 3,5
    7. Assume NOT(C)
    8. P AND NOT(C) proof of conjunction 1,7
    9. (P AND C) OR (P AND NOT(C)) proof of disjunction 8
10. NOT(C) IMPLIES ((P AND C) OR (P AND NOT(C))) direct proof 7,9
11. (P AND C) OR (P AND NOT(C)) proof by cases 2,6,10
12. P IMPLIES ((P AND C) OR (P AND NOT(C))) direct proof 1,11

LEMMA: P IMPLIES ((P AND C) OR (P AND NOT(C)))

1. Assume P
2. C OR NOT C tautology
    3. Assume C
    4. P AND C proof of conjunction 1,3
    5. (P AND C) OR (P AND NOT(C)) proof of disjunction 4
6. C IMPLIES ((P AND C) OR (P AND NOT(C))) direct proof 3,5
    7. Assume NOT(C)
    8. P AND NOT(C) proof of conjunction 1,7
    9. (P AND C) OR (P AND NOT(C)) proof of disjunction 8
10. NOT(C) IMPLIES ((P AND C) OR (P AND NOT(C))) direct proof 7,9
11. (P AND C) OR (P AND NOT(C)) proof by cases 2,6,10

12. P IMPLIES ((P AND C) OR (P AND NOT(C))) direct proof 1,11

LEMMA: P IMPLIES ((P AND C) OR (P AND NOT(C)))

1. Assume P
2. C OR NOT C tautology
    3. Assume C
    4. P AND C proof of conjunction 1,3
    5. (P AND C) OR (P AND NOT(C)) proof of disjunction 4
6. C IMPLIES ((P AND C) OR (P AND NOT(C))) direct proof 3,5
    7. Assume NOT(C)
    8. P AND NOT(C) proof of conjunction 1,7
    9. (P AND C) OR (P AND NOT(C)) proof of disjunction 8
10. NOT(C) IMPLIES ((P AND C) OR (P AND NOT(C))) direct proof 7,9
11. (P AND C) OR (P AND NOT(C)) proof by cases 2,6,10

12. P IMPLIES ((P AND C) OR (P AND NOT(C))) direct proof 1,11

LEMMA: P IMPLIES ((P AND C) OR (P AND NOT(C)))

1. Assume P
2. C OR NOT C tautology
    3. Assume C
    4. P AND C proof of conjunction 1,3
    5. (P AND C) OR (P AND NOT(C)) proof of disjunction 4
6. C IMPLIES ((P AND C) OR (P AND NOT(C))) direct proof 3,5
    7. Assume NOT(C)
    8. P AND NOT(C) proof of conjunction 1,7
    9. (P AND C) OR (P AND NOT(C)) proof of disjunction 8
10. NOT(C) IMPLIES ((P AND C) OR (P AND NOT(C))) direct proof 7,9
11. (P AND C) OR (P AND NOT(C)) proof by cases 2,6,10

12. P IMPLIES ((P AND C) OR (P AND NOT(C))) direct proof 1,11

LEMMA: P IMPLIES ((P AND C) OR (P AND NOT(C)))

1. Assume P
2. C OR NOT C tautology
    3. Assume C
    4. P AND C proof of conjunction 1,3
    5. (P AND C) OR (P AND NOT(C)) proof of disjunction 4
6. C IMPLIES ((P AND C) OR (P AND NOT(C))) direct proof 3,5
    7. Assume NOT(C)
    8. P AND NOT(C) proof of conjunction 1,7
    9. (P AND C) OR (P AND NOT(C)) proof of disjunction 8
10. NOT(C) IMPLIES ((P AND C) OR (P AND NOT(C))) direct proof 7,9
11. (P AND C) OR (P AND NOT(C)) proof by cases 2,6,10
12. P IMPLIES ((P AND C) OR (P AND NOT(C))) direct proof 1,11

LEMMA: P IMPLIES ((P AND C) OR (P AND NOT(C)))

1. Assume P
2. C OR NOT C tautology
    3. Assume C
    4. P AND C proof of conjunction 1,3
    5. (P AND C) OR (P AND NOT(C)) proof of disjunction 4
6. C IMPLIES ((P AND C) OR (P AND NOT(C))) direct proof 3,5
    7. Assume NOT(C)
    8. P AND NOT(C) proof of conjunction 1,7
    9. (P AND C) OR (P AND NOT(C)) proof of disjunction 8
10. NOT(C) IMPLIES ((P AND C) OR (P AND NOT(C))) direct proof 7,9
11. (P AND C) OR (P AND NOT(C)) proof by cases 2,6,10

12. P IMPLIES ((P AND C) OR (P AND NOT(C))) direct proof 1,11

LEMMA: P IMPLIES ((P AND C) OR (P AND NOT(C)))

1. Assume P
2. C OR NOT C tautology
    3. Assume C
    4. P AND C proof of conjunction 1,3
    5. (P AND C) OR (P AND NOT(C)) proof of disjunction 4
6. C IMPLIES ((P AND C) OR (P AND NOT(C))) direct proof 3,5
    7. Assume NOT(C)
    8. P AND NOT(C) proof of conjunction 1,7
    9. (P AND C) OR (P AND NOT(C)) proof of disjunction 8
10. NOT(C) IMPLIES ((P AND C) OR (P AND NOT(C))) direct proof 7,9
11. (P AND C) OR (P AND NOT(C)) proof by cases 2,6,10

12. P IMPLIES ((P AND C) OR (P AND NOT(C))) direct proof 1,11

LEMMA: P IMPLIES ((P AND C) OR (P AND NOT(C)))

1. Assume P
2. C OR NOT C tautology
    3. Assume C
    4. P AND C proof of conjunction 1,3
    5. (P AND C) OR (P AND NOT(C)) proof of disjunction 4
6. C IMPLIES ((P AND C) OR (P AND NOT(C))) direct proof 3,5
    7. Assume NOT(C)
    8. P AND NOT(C) proof of conjunction 1,7
    9. (P AND C) OR (P AND NOT(C)) proof of disjunction 8
10. NOT(C) IMPLIES ((P AND C) OR (P AND NOT(C))) direct proof 7,9
11. (P AND C) OR (P AND NOT(C)) proof by cases 2,6,10
12. P IMPLIES ((P AND C) OR (P AND NOT(C))) direct proof 1,11

LEMMA: P IMPLIES ((P AND C) OR (P AND NOT(C)))

1. Assume P
2. C OR NOT C tautology
    3. Assume C
    4. P AND C proof of conjunction 1,3
    5. (P AND C) OR (P AND NOT(C)) proof of disjunction 4
6. C IMPLIES ((P AND C) OR (P AND NOT(C))) direct proof 3,5
    7. Assume NOT(C)
    8. P AND NOT(C) proof of conjunction 1,7
    9. (P AND C) OR (P AND NOT(C)) proof of disjunction 8
10. NOT(C) IMPLIES ((P AND C) OR (P AND NOT(C))) direct proof 7,9
11. (P AND C) OR (P AND NOT(C)) proof by cases 2,6,10
12. P IMPLIES ((P AND C) OR (P AND NOT(C))) direct proof 1,11

LEMMA: P IMPLIES ((P AND C) OR (P AND NOT(C)))

1. Assume P
2. C OR NOT C tautology
   3. Assume C
   4. P AND C proof of conjunction 1,3
   5. (P AND C) OR (P AND NOT(C)) proof of disjunction 4
6. C IMPLIES ((P AND C) OR (P AND NOT(C))) direct proof 3,5
   7. Assume NOT(C)
   8. P AND NOT(C) proof of conjunction 1,7
   9. (P AND C) OR (P AND NOT(C)) proof of disjunction 8
10. NOT(C) IMPLIES ((P AND C) OR (P AND NOT(C))) direct proof 7,9
11. (P AND C) OR (P AND NOT(C)) proof by cases 2,6,10
12. P IMPLIES ((P AND C) OR (P AND NOT(C))) direct proof 1,11

LEMMA: ((P AND C) OR (P AND NOT(C))) IMPLIES P

1. Assume P AND C
2. P use of conjunction 1

3. (P AND C) IMPLIES P direct proof 1,2

4. Assume P AND NOT(C)
5. P use of conjunction 5

6. (P AND NOT(C)) IMPLIES P direct proof 4,5
7. ((P AND C) OR (P AND NOT(C))) IMPLIES P proof by cases 3,6

LEMMA: $\lfloor (n+1)/2 \rfloor = \lceil n/2 \rceil$.

Proof:

Suppose n is even. Then $n = 2k$ for some integer k, so
$\lfloor (n+1)/2 \rfloor = \lfloor (2k+1)/2 \rfloor = \lfloor k + 1/2 \rfloor = k$
$= \lceil k \rceil = \lceil 2k/2 \rceil = \lceil n/2 \rceil$.
Then n is even IMPLIES $\lfloor (n+1)/2 \rfloor = \lceil n/2 \rceil$.

Suppose n is odd. Then $n = 2k+1$ for some integer k, so
$\lfloor (n+1)/2 \rfloor = \lfloor (2k+2)/2 \rfloor = \lfloor k+1 \rfloor = k+1$
$= \lceil k + 1/2 \rceil = \lceil (2k+1)/2 \rceil = \lceil n/2 \rceil$.
Then n is odd IMPLIES $\lfloor (n+1)/2 \rfloor = \lceil n/2 \rceil$.

Since n is even or n is odd, it follows that
$\lfloor (n+1)/2 \rfloor = \lceil n/2 \rceil$. ■

Let $x \in D$

$\vdots$

$p(x)$

Since x is an arbitrary element of D,

$\forall x \in D. p(x)$

LEMMA: For all integers x, if x is even, then x^2 is even.

Proof:

Let x be an integer.

Assume x is even.

Then $x = 2k$ for some integer k,

so $x^2 = (2k)^2 = 2 \times (2k^2)$, which is even.

Hence, x is even implies that x^2 is even.

Since x is an arbitrary integer,

for all integers x, if x is even, then x^2 is even. ∎

LEMMA: For all integers x, if x is even, then x^2 is even.

Proof:

Let x be an integer.

Assume x is even.

Then $x = 2k$ for some integer k,

so $x^2 = (2k)^2 = 2 \times (2k^2)$, which is even.

Hence, x is even implies that x^2 is even.

Since x is an arbitrary integer,

for all integers x, if x is even, then x^2 is even. ∎

LEMMA: For all integers x, if x is even, then x^2 is even.

Proof:

Let x be an integer.

Assume x is even.

Then $x = 2k$ for some integer k,

so $x^2 = (2k)^2 = 2 \times (2k^2)$, which is even.

Hence, x is even implies that x^2 is even.

Since x is an arbitrary integer,

for all integers x, if x is even, then x^2 is even. ∎

LEMMA: For all integers x, if x is even, then x^2 is even.

Proof:

Let x be an integer.

Assume x is even.

Then $x = 2k$ for some integer k,

so $x^2 = (2k)^2 = 2 \times (2k^2)$, which is even.

Hence, x is even implies that x^2 is even.

Since x is an arbitrary integer,

for all integers x, if x is even, then x^2 is even. ■

LEMMA: For all integers x, if x is even, then x^2 is even.

Proof:

Let x be an integer.

Assume x is even.

Then $x = 2k$ for some integer k,

so $x^2 = (2k)^2 = 2 \times (2k^2)$, which is even.

Hence, x is even implies that x^2 is even.

Since x is an arbitrary integer,

for all integers x, if x is even, then x^2 is even. ∎

LEMMA: For all integers x, if x is even, then x^2 is even.

Proof:

Let x be an integer.

Assume x is even.

Then $x = 2k$ for some integer k,

so $x^2 = (2k)^2 = 2 \times (2k^2)$, which is even.

Hence, x is even implies that x^2 is even.

Since x is an arbitrary integer,

for all integers x, if x is even, then x^2 is even. ∎

LEMMA: For all even integers x, x^2 is even.

Proof:

Let x be an even integer.

Then $x = 2k$ for some integer k,

so $x^2 = (2k)^2 = 2 \times (2k^2)$, which is even.

Since x is an arbitrary integer,

for all even integers x, x^2 is even. ■

Construction

Let $x =$ ____

$\vdots$

Therefore $x \in D$.

$\vdots$

$p(x)$

$\exists x \in D . p(x)$

Instantiation

$\exists x \in D.p(x)$

Let $y \in D$ be such that $p(y)$.

$\vdots$

For any integer y, let $largest(y) = \text{NOT}(\exists x \in \mathbb{Z}.(x > y))$

THEOREM: $\text{NOT}(\exists y \in \mathbb{Z}.largest(y))$

Proof:

1. Assume $\exists y \in \mathbb{Z}.largest(y)$.
2. Let y be an integer that satisfies $largest(y)$. instantiation 1
3. $\text{NOT}(\exists x \in \mathbb{Z}.(x > y))$ definition 2
    4. Let $x = y + 1$.
    5. $\forall z \in \mathbb{Z}.(z + 1 \text{ is an integer})$ axiom.
    6. $x = y + 1$ is an integer. specialization 5
    7. $\forall z \in \mathbb{Z}.(z + 1 > z)$ axiom.
    8. $x = y + 1 > y$ specialization 7
9. $\exists x \in \mathbb{Z}.(x > y)$ construction 6,8
10. $\text{NOT}(\exists y \in \mathbb{Z}.largest(y))$ proof by contradiction 1,3,9

For any integer y, let $largest(y) = \text{NOT}(\exists x \in \mathbb{Z}.(x > y))$

THEOREM: $\text{NOT}(\exists y \in \mathbb{Z}.largest(y))$

Proof:

1. Assume $\exists y \in \mathbb{Z}.largest(y)$.
2. Let y be an integer that satisfies $largest(y)$. instantiation 1
3. $\text{NOT}(\exists x \in \mathbb{Z}.(x > y))$ definition 2
   4. Let $x = y + 1$.
   5. $\forall z \in \mathbb{Z}.(z + 1$ is an integer) axiom.
   6. $x = y + 1$ is an integer. specialization 5
   7. $\forall z \in \mathbb{Z}.(z + 1 > z)$ axiom.
   8. $x = y + 1 > y$ specialization 7
9. $\exists x \in \mathbb{Z}.(x > y)$ construction 6,8
10. $\text{NOT}(\exists y \in \mathbb{Z}.largest(y))$ proof by contradiction 1,3,9

For any integer y, let $largest(y) = \text{NOT}(\exists x \in \mathbb{Z}.(x > y))$

THEOREM: $\text{NOT}(\exists y \in \mathbb{Z}.largest(y))$

Proof:

1. Assume $\exists y \in \mathbb{Z}.largest(y)$.
2. Let y be an integer that satisfies $largest(y)$. instantiation 1
3. $\text{NOT}(\exists x \in \mathbb{Z}.(x > y))$ definition 2
    4. Let $x = y + 1$.
    5. $\forall z \in \mathbb{Z}.(z + 1 \text{ is an integer})$ axiom.
    6. $x = y + 1$ is an integer. specialization 5
    7. $\forall z \in \mathbb{Z}.(z + 1 > z)$ axiom.
    8. $x = y + 1 > y$ specialization 7
9. $\exists x \in \mathbb{Z}.(x > y)$ construction 6,8
10. $\text{NOT}(\exists y \in \mathbb{Z}.largest(y))$ proof by contradiction 1,3,9

For any integer y, let $\mathit{largest}(y) = \text{NOT}(\exists x \in \mathbb{Z}.(x > y))$

THEOREM: $\text{NOT}(\exists y \in \mathbb{Z}.\mathit{largest}(y))$

Proof:

1. Assume $\exists y \in \mathbb{Z}.\mathit{largest}(y)$.
2. Let y be an integer that satisfies $\mathit{largest}(y)$. instantiation 1
3. $\text{NOT}(\exists x \in \mathbb{Z}.(x > y))$ definition 2
   4. Let $x = y + 1$.
   5. $\forall z \in \mathbb{Z}.(z + 1 \text{ is an integer})$ axiom.
   6. $x = y + 1$ is an integer. specialization 5
   7. $\forall z \in \mathbb{Z}.(z + 1 > z)$ axiom.
   8. $x = y + 1 > y$ specialization 7
9. $\exists x \in \mathbb{Z}.(x > y)$ construction 6,8
10. $\text{NOT}(\exists y \in \mathbb{Z}.\mathit{largest}(y))$ proof by contradiction 1,3,9

For any integer y, let $largest(y) = \text{NOT}(\exists x \in \mathbb{Z}.(x > y))$

THEOREM: $\text{NOT}(\exists y \in \mathbb{Z}.largest(y))$

Proof:

1. Assume $\exists y \in \mathbb{Z}.largest(y)$.
2. Let y be an integer that satisfies $largest(y)$. instantiation 1
3. $\text{NOT}(\exists x \in \mathbb{Z}.(x > y))$ definition 2
    4. Let $x = y + 1$.
    5. $\forall z \in \mathbb{Z}.(z + 1 \text{ is an integer})$ axiom.
    6. $x = y + 1$ is an integer. specialization 5
    7. $\forall z \in \mathbb{Z}.(z + 1 > z)$ axiom.
    8. $x = y + 1 > y$ specialization 7
9. $\exists x \in \mathbb{Z}.(x > y)$ construction 6,8
10. $\text{NOT}(\exists y \in \mathbb{Z}.largest(y))$ proof by contradiction 1,3,9

For any integer y, let $\mathit{largest}(y) = \text{NOT}(\exists x \in \mathbb{Z}.(x > y))$

THEOREM: $\text{NOT}(\exists y \in \mathbb{Z}.\mathit{largest}(y))$

Proof:

1. Assume $\exists y \in \mathbb{Z}.\mathit{largest}(y)$.
2. Let y be an integer that satisfies $\mathit{largest}(y)$. instantiation 1
3. $\text{NOT}(\exists x \in \mathbb{Z}.(x > y))$ definition 2
    4. Let $x = y + 1$.
    5. $\forall z \in \mathbb{Z}.(z + 1$ is an integer) axiom.
    6. $x = y + 1$ is an integer. specialization 5
    7. $\forall z \in \mathbb{Z}.(z + 1 > z)$ axiom.
    8. $x = y + 1 > y$ specialization 7
9. $\exists x \in \mathbb{Z}.(x > y)$ construction 6,8
10. $\text{NOT}(\exists y \in \mathbb{Z}.\mathit{largest}(y))$ proof by contradiction 1,3,9

For any integer y, let $largest(y) = \text{NOT}(\exists x \in \mathbb{Z}.(x > y))$

THEOREM: $\text{NOT}(\exists y \in \mathbb{Z}.largest(y))$

Proof:

1. Assume $\exists y \in \mathbb{Z}.largest(y)$.
2. Let y be an integer that satisfies $largest(y)$. instantiation 1
3. $\text{NOT}(\exists x \in \mathbb{Z}.(x > y))$ definition 2
    4. Let $x = y + 1$.
    5. $\forall z \in \mathbb{Z}.(z + 1 \text{ is an integer})$ axiom.
    6. $x = y + 1$ is an integer. specialization 5
    7. $\forall z \in \mathbb{Z}.(z + 1 > z)$ axiom.
    8. $x = y + 1 > y$ specialization 7
9. $\exists x \in \mathbb{Z}.(x > y)$ construction 6,8
10. $\text{NOT}(\exists y \in \mathbb{Z}.largest(y))$ proof by contradiction 1,3,9

For any integer y, let $largest(y) = \text{NOT}(\exists x \in \mathbb{Z}.(x > y))$

THEOREM: $\text{NOT}(\exists y \in \mathbb{Z}.largest(y))$

Proof:

1. Assume $\exists y \in \mathbb{Z}.largest(y)$.
2. Let y be an integer that satisfies $largest(y)$. instantiation 1
3. $\text{NOT}(\exists x \in \mathbb{Z}.(x > y))$ definition 2
    4. Let $x = y + 1$.
    5. $\forall z \in \mathbb{Z}.(z + 1 \text{ is an integer})$ axiom.
    6. $x = y + 1$ is an integer. specialization 5
    7. $\forall z \in \mathbb{Z}.(z + 1 > z)$ axiom.
    8. $x = y + 1 > y$ specialization 7
9. $\exists x \in \mathbb{Z}.(x > y)$ construction 6,8

10. $\text{NOT}(\exists y \in \mathbb{Z}.largest(y))$ proof by contradiction 1,3,9

For any integer y, let $largest(y) = \text{NOT}(\exists x \in \mathbb{Z}.(x > y))$

THEOREM: $\text{NOT}(\exists y \in \mathbb{Z}.largest(y))$

Proof:

1. Assume $\exists y \in \mathbb{Z}.largest(y)$.
2. Let y be an integer that satisfies $largest(y)$. instantiation 1
3. $\text{NOT}(\exists x \in \mathbb{Z}.(x > y))$ definition 2
    4. Let $x = y + 1$.
    5. $\forall z \in \mathbb{Z}.(z + 1 \text{ is an integer})$ axiom.
    6. $x = y + 1$ is an integer. specialization 5
    7. $\forall z \in \mathbb{Z}.(z + 1 > z)$ axiom.
    8. $x = y + 1 > y$ specialization 7
9. $\exists x \in \mathbb{Z}.(x > y)$ construction 6,8
10. $\text{NOT}(\exists y \in \mathbb{Z}.largest(y))$ proof by contradiction 1,3,9

Additional Reading

- Proof Outlines
- Mathematics for Computer Science, Section 1.9
- Writing Mathematics
- How to Write a 21st Century Proof
- Generalized Logic
- Proof techniques you shouldn't use

Problem 1

(a) Why is the following proof of $1/8 > 1/4$ bogus?

$$
\begin{array}{rcl}
3 & > & 2 \\
3\log_{10}(1/2) & > & 2\log_{10}(1/2) \\
\log_{10}(1/2)^3 & > & \log_{10}(1/2)^2 \\
1/8 = (1/2)^3 & > & (1/2)^2 = 1/4
\end{array}
$$

(b) Why is the following proof bogus? It claims to prove that if a and b are equal real numbers, then $a = 0$.

$$
\begin{array}{rcl}
a & = & b \\
a^2 & = & ab \\
a^2 - b^2 & = & ab - b^2 \\
(a-b)(a+b) & = & (a-b)b \\
a+b & = & b \\
a & = & 0
\end{array}
$$

Problem 1

(a) Why is the following proof of $1/8 > 1/4$ bogus?

$$\begin{aligned} 3 &> 2 \\ 3\log_{10}(1/2) &> 2\log_{10}(1/2) \\ \log_{10}(1/2)^3 &> \log_{10}(1/2)^2 \\ 1/8 = (1/2)^3 &> (1/2)^2 = 1/4 \end{aligned}$$

$\log_{10} x < 0$ for $0 < x < 1$.
Since the second line is obtained from the first by multiplying both sides by a negative number, the inequality needs to be reversed.

Problem 1

(b) Why is the following proof bogus? It claims to prove that if a and b are equal real numbers, then $a = 0$.

$$
\begin{aligned}
a &= b \\
a^2 &= ab \\
a^2 - b^2 &= ab - b^2 \\
(a-b)(a+b) &= (a-b)b \\
a+b &= b \\
a &= 0
\end{aligned}
$$

Since $a - b = 0$, you can't divide both sides of the equation on line 4 to get the equation on line 5.

Problem 2

Formally prove:
((A IMPLIES B) AND (NOT(A) IMPLIES B)) IMPLIES B

10. ((A IMPLIES B) AND (NOT(A) IMPLIES B)) IMPLIES B

1. (A IMPLIES B) AND (NOT(A) IMPLIES B) assumption

9. B
10. ((A IMPLIES B) AND (NOT(A) IMPLIES B)) IMPLIES B
    direct proof 1,9

1. (A IMPLIES B) AND (NOT(A) IMPLIES B) assumption
2. Assume NOT(B)

5. NOT(A)

8. A
9. B proof by contradiction 2,5,8
10. ((A IMPLIES B) AND (NOT(A) IMPLIES B)) IMPLIES B direct proof 1,9

1. (A IMPLIES B) AND (NOT(A) IMPLIES B) assumption
   2. Assume NOT(B)
   3. A IMPLIES B use of conjunction 1

   5. NOT(A)

   8. A
9. B proof by contradiction 2,5,8
10. ((A IMPLIES B) AND (NOT(A) IMPLIES B)) IMPLIES B
    direct proof 1,9

1. (A IMPLIES B) AND (NOT(A) IMPLIES B) assumption
   2. Assume NOT(B)
   3. A IMPLIES B use of conjunction 1
   4. NOT(B) IMPLIES NOT(A) contrapositive 3
   5. NOT(A)
   8. A
9. B proof by contradiction 2,5,8
10. ((A IMPLIES B) AND (NOT(A) IMPLIES B)) IMPLIES B direct proof 1,9

1. (A IMPLIES B) AND (NOT(A) IMPLIES B) assumption
   2. Assume NOT(B)
   3. A IMPLIES B use of conjunction 1
   4. NOT(B) IMPLIES NOT(A) contrapositive 3
   5. NOT(A) modus ponens 2,4

   8. A
9. B proof by contradiction 2,5,8
10. ((A IMPLIES B) AND (NOT(A) IMPLIES B)) IMPLIES B direct proof 1,9

1. (A IMPLIES B) AND (NOT(A) IMPLIES B) assumption
   2. Assume NOT(B)
   3. A IMPLIES B use of conjunction 1
   4. NOT(B) IMPLIES NOT(A) contrapositive 3
   5. NOT(A) modus ponens 2,4
   6. NOT(A) IMPLIES B use of conjunction 1
   8. A
9. B proof by contradiction 2,5,8
10. ((A IMPLIES B) AND (NOT(A) IMPLIES B)) IMPLIES B direct proof 1,9

1. (A IMPLIES B) AND (NOT(A) IMPLIES B) assumption
2. Assume NOT(B)
3. A IMPLIES B use of conjunction 1
4. NOT(B) IMPLIES NOT(A) contrapositive 3
5. NOT(A) modus ponens 2,4
6. NOT(A) IMPLIES B use of conjunction 1
7. NOT(B) IMPLIES A contrapositive 6
8. A
9. B proof by contradiction 2,5,8
10. ((A IMPLIES B) AND (NOT(A) IMPLIES B)) IMPLIES B direct proof 1,9

1. (A IMPLIES B) AND (NOT(A) IMPLIES B) assumption
   2. Assume NOT(B)
   3. A IMPLIES B use of conjunction 1
   4. NOT(B) IMPLIES NOT(A) contrapositive 3
   5. NOT(A) modus ponens 2,4
   6. NOT(A) IMPLIES B use of conjunction 1
   7. NOT(B) IMPLIES A contrapositive 6
   8. A modus ponens 2,7
9. B proof by contradiction 2,5,8
10. ((A IMPLIES B) AND (NOT(A) IMPLIES B)) IMPLIES B direct proof 1,9

1. (A IMPLIES B) AND (NOT(A) IMPLIES B) assumption
    2. Assume NOT(B)
    3. A IMPLIES B use of conjunction 1
    4. NOT(B) IMPLIES NOT(A) contrapositive 3
    5. NOT(A) modus ponens 2,4
    6. NOT(A) IMPLIES B use of conjunction 1
    7. NOT(B) IMPLIES A contrapositive 6
    8. A modus ponens 2,7
9. B proof by contradiction 2,5,8
10. ((A IMPLIES B) AND (NOT(A) IMPLIES B)) IMPLIES B direct proof 1,9

Note that we indent at lines 1 and 2, when we make assumptions.

Problem 3

Suppose that:
R is logically equivalent to S,
R' is obtained from R by making a substitution,
S' is obtained from S by making the same substitution,
and R' is true.

Prove S' is true or give a counterexample.

Let
R = (P IMPLIES Q) IMPLIES P,
S = P AND P,
R' = (P IMPLIES Q) IMPLIES U, and
S' = P AND U.

R is logically equivalent to S (check their truth tables),
R' is obtained from R by substituting U for the second P
S' is obtained from S by substituting U for the second P
and, when P = T and Q = U = F, $R' = T$ but $S' = F$.

Let
$R =$ (P IMPLIES Q) IMPLIES P,
$S =$ P AND P,
$R' =$ (P IMPLIES Q) IMPLIES U, and
$S' =$ P AND U.

R is logically equivalent to S (check their truth tables),
R' is obtained from R by substituting U for the second P
S' is obtained from S by substituting U for the second P
and, when P = T and Q = U = F, $R' =$ T but $S' =$ F.

Not all occurrences of P in R are replaced by U.

Let
R = (P OR Q) IMPLIES (P AND Q),
S = P IFF (Q OR (P AND Q)),
R' = (P OR Q) IMPLIES U, and
S' = P IFF (Q OR U).

R is logically equivalent to S (check their truth tables),
R' is obtained from R by substituting U for (P AND Q)
S' is obtained from S by substituting U for (P AND Q)
and, when P = Q = F and U = T, R' = T but S' = F.

Let
R = (P OR Q) IMPLIES (P AND Q),
S = P IFF (Q OR (P AND Q)),
R' = (P OR Q) IMPLIES U, and
S' = P IFF (Q OR U).

R is logically equivalent to S (check their truth tables),
R' is obtained from R by substituting U for (P AND Q)
S' is obtained from S by substituting U for (P AND Q)
and, when P = Q = F and U = T, R' = T but S' = F.

Substituting for a formula, rather than a variable.

Suppose that:
R is logically equivalent to S,
R' is obtained from R by making a substitution,
S' is obtained from S by making the same substitution,
and R' is true.

Prove S' is true under more precise conditions.

Suppose that:

R is logically equivalent to S,

R' is obtained from R by substituting all occurrences of some propositional variable P by a formula Q,

S' is obtained from S by making the same substitution,

and R' is true.

Then S is true.

Proof:

1. R' assumption
2. R IFF S assumption
3. R IMPLIES S use of equivalence 2
4. R' IMPLIES S' substitution 3 (see lecture 5 slide 2)
5. S' modus ponens 1,4

Problem 4

Give a formal proof of:
$[\forall x \in D.(R(x) \text{ IFF } \forall y \in D.A(x,y))]$ IMPLIES
$[\exists x \in D.\forall y \in D.(R(y) \text{ IMPLIES } A(x,y))]$.
Use a top down approach.

26 $[\forall x \in D.(R(x) \text{ IFF } \forall y \in D.A(x,y))]$ IMPLIES
$[\exists x \in D.\forall y \in D.(R(y) \text{ IMPLIES } A(x,y))]$

1 Assume $\forall x \in D.(R(x) \text{ IFF } \forall y \in D.A(x,y))$

25 $\exists x \in D.\forall y \in D.(R(y) \text{ IMPLIES } A(x,y))$

26 $[\forall x \in D.(R(x) \text{ IFF } \forall y \in D.A(x,y))]$ IMPLIES
$[\exists x \in D.\forall y \in D.(R(y) \text{ IMPLIES } A(x,y))]$ direct proof 1,25

1 Assume $\forall x \in D.(R(x) \text{ IFF } \forall y \in D.A(x,y))$

3 $(\exists x \in D.R(x))$ OR NOT$(\exists x \in D.R(x))$

14 $[\exists x \in D.R(x)]$ IMPLIES $[\exists x \in D.\forall y \in D.(R(y) \text{ IMPLIES } A(x,y))]$

24 [NOT$(\exists x \in D.R(x))$] IMPLIES
$[\exists x \in D.\forall y \in D.(R(y) \text{ IMPLIES } A(x,y))]$

25 $\exists x \in D.\forall y \in D.(R(y) \text{ IMPLIES } A(x,y))$ proof by cases 3, 14, 24

26 $[\forall x \in D.(R(x) \text{ IFF } \forall y \in D.A(x,y))]$ IMPLIES
$[\exists x \in D.\forall y \in D.(R(y) \text{ IMPLIES } A(x,y))]$ direct proof 1,25

1 Assume $\forall x \in D.(R(x) \text{ IFF } \forall y \in D.A(x,y))$

2 P OR NOT(P) tautology

3 $(\exists x \in D.R(x))$ OR NOT$(\exists x \in D.R(x))$ substitution 2

14 $[\exists x \in D.R(x)]$ IMPLIES $[\exists x \in D.\forall y \in D.(R(y) \text{ IMPLIES } A(x,y))]$

24 [NOT$(\exists x \in D.R(x))$] IMPLIES
$[\exists x \in D.\forall y \in D.(R(y) \text{ IMPLIES } A(x,y))]$

25 $\exists x \in D.\forall y \in D.(R(y) \text{ IMPLIES } A(x,y))$ proof by cases 3, 14, 24

26 $[\forall x \in D.(R(x) \text{ IFF } \forall y \in D.A(x,y))]$ IMPLIES
$[\exists x \in D.\forall y \in D.(R(y) \text{ IMPLIES } A(x,y))]$ direct proof 1,25

1 Assume $\forall x \in D.(R(x) \text{ IFF } \forall y \in D.A(x,y))$

2 P OR NOT(P) tautology

3 $(\exists x \in D.R(x))$ OR NOT$(\exists x \in D.R(x))$ substitution 2

4 Suppose $\exists x \in D.R(x)$

13 $\exists x \in D.\forall y \in D.(R(y) \text{ IMPLIES } A(x,y))$

14 $[\exists x \in D.R(x)]$ IMPLIES $[\exists x \in D.\forall y \in D.(R(y) \text{ IMPLIES } A(x,y))]$
direct proof 4,13

24 [NOT$(\exists x \in D.R(x))$] IMPLIES
$[\exists x \in D.\forall y \in D.(R(y) \text{ IMPLIES } A(x,y))]$

25 $\exists x \in D.\forall y \in D.(R(y) \text{ IMPLIES } A(x,y))$ proof by cases 3, 14, 24

26 $[\forall x \in D.(R(x) \text{ IFF } \forall y \in D.A(x,y))]$ IMPLIES
$[\exists x \in D.\forall y \in D.(R(y) \text{ IMPLIES } A(x,y))]$ direct proof 1,25

1 Assume $\forall x \in D.(R(x) \text{ IFF } \forall y \in D.A(x,y))$

2 P OR NOT(P) tautology

3 $(\exists x \in D.R(x))$ OR NOT$(\exists x \in D.R(x))$ substitution 2

4 Suppose $\exists x \in D.R(x)$

13 $\exists x \in D.\forall y \in D.(R(y) \text{ IMPLIES } A(x,y))$

14 $[\exists x \in D.R(x)]$ IMPLIES $[\exists x \in D.\forall y \in D.(R(y) \text{ IMPLIES } A(x,y))]$
direct proof 4,13

15 Suppose NOT$(\exists x \in D.R(x))$

23 $\exists x \in D.\forall y \in D.(R(y) \text{ IMPLIES } A(x,y))$

24 [NOT$(\exists x \in D.R(x))$] IMPLIES
$[\exists x \in D.\forall y \in D.(R(y) \text{ IMPLIES } A(x,y))]$ direct proof 15, 23

25 $\exists x \in D.\forall y \in D.(R(y) \text{ IMPLIES } A(x,y))$ proof by cases 3, 14, 24

26 $[\forall x \in D.(R(x) \text{ IFF } \forall y \in D.A(x,y))]$ IMPLIES
$[\exists x \in D.\forall y \in D.(R(y) \text{ IMPLIES } A(x,y))]$ direct proof 1,25

1 Assume $\forall x \in D.(R(x) \text{ IFF } \forall y \in D.A(x, y))$

4 Suppose $\exists x \in D.R(x)$

13 $\exists x \in D.\forall y \in D.(R(y) \text{ IMPLIES } A(x, y))$

1 Assume $\forall x \in D.(R(x) \text{ IFF } \forall y \in D.A(x, y))$

4 Suppose $\exists x \in D.R(x)$

5 Let $d \in D$ be such that $R(d)$ instantiation 4

13 $\exists x \in D.\forall y \in D.(R(y) \text{ IMPLIES } A(x, y))$

1 Assume $\forall x \in D.(R(x) \text{ IFF } \forall y \in D.A(x,y))$

4 Suppose $\exists x \in D.R(x)$

5 Let $d \in D$ be such that $R(d)$ instantiation 4

6 $R(d)$ IFF $\forall y \in D.A(d,y)$ specialization 1

13 $\exists x \in D.\forall y \in D.(R(y) \text{ IMPLIES } A(x,y))$

1 Assume $\forall x \in D.(R(x) \text{ IFF } \forall y \in D.A(x, y))$

4 Suppose $\exists x \in D.R(x)$

5 Let $d \in D$ be such that $R(d)$ instantiation 4

6 $R(d)$ IFF $\forall y \in D.A(d, y)$, specialization 1

7 $\forall y \in D.A(d, y)$ modus ponens 6 5

13 $\exists x \in D.\forall y \in D.(R(y) \text{ IMPLIES } A(x, y))$

1 Assume $\forall x \in D.(R(x) \text{ IFF } \forall y \in D.A(x, y))$

4 Suppose $\exists x \in D.R(x)$

5 Let $d \in D$ be such that $R(d)$ instantiation 4

6 $R(d)$ IFF $\forall y \in D.A(d, y)$, specialization 1

7 $\forall y \in D.A(d, y)$ modus ponens 6,5

12 $\forall y \in D.(R(y) \text{ IMPLIES } A(d, y))$

13 $\exists x \in D.\forall y \in D.(R(y) \text{ IMPLIES } A(x, y))$ construction 5,12

1 Assume $\forall x \in D.(R(x) \text{ IFF } \forall y \in D.A(x, y))$

4 Suppose $\exists x \in D.R(x)$

5 Let $d \in D$ be such that $R(d)$ instantiation 4

6 $R(d)$ IFF $\forall y \in D.A(d, y)$, specialization 1

7 $\forall y \in D.A(d, y)$ modus ponens 6,5

8 Let $y \in D$ be arbitrary.

11 $R(y)$ IMPLIES $A(d, y)$

12 $\forall y \in D.(R(y) \text{ IMPLIES } A(d, y))$ generalization 8,11

13 $\exists x \in D.\forall y \in D.(R(y) \text{ IMPLIES } A(x, y))$ construction 5,12

1 Assume $\forall x \in D.(R(x) \text{ IFF } \forall y \in D.A(x,y))$

4 Suppose $\exists x \in D.R(x)$

5 Let $d \in D$ be such that $R(d)$ instantiation 4

6 $R(d)$ IFF $\forall y \in D.A(d,y)$, specialization 1

7 $\forall y \in D.A(d,y)$ modus ponens 6,5

8 Let $y \in D$ be arbitrary.

9 Assume $R(y)$.

10 $A(d,y)$

11 $R(y)$ IMPLIES $A(d,y)$ direct proof of implication 9,10

12 $\forall y \in D.(R(y) \text{ IMPLIES } A(d,y))$ generalization 8,11

13 $\exists x \in D.\forall y \in D.(R(y) \text{ IMPLIES } A(x,y))$ construction 5,12

1 Assume $\forall x \in D.(R(x) \text{ IFF } \forall y \in D.A(x, y))$

4 Suppose $\exists x \in D.R(x)$

5 Let $d \in D$ be such that $R(d)$ instantiation 4

6 $R(d)$ IFF $\forall y \in D.A(d, y)$, specialization 1

7 $\forall y \in D.A(d, y)$ modus ponens 6,5

8 Let $y \in D$ be arbitrary.

9 Assume $R(y)$.

10 $A(d, y)$ specialization 7

11 $R(y)$ IMPLIES $A(d, y)$ direct proof of implication 9,10

12 $\forall y \in D.(R(y) \text{ IMPLIES } A(d, y))$ generalization 8,11

13 $\exists x \in D.\forall y \in D.(R(y) \text{ IMPLIES } A(x, y))$ construction 5,12

15 Suppose NOT$(\exists x \in D. R(x))$

23 $\exists x \in D. \forall y \in D. (R(y)$ IMPLIES $A(x, y))$

15 Suppose NOT($\exists x \in D.R(x)$)

16 $\forall x \in D.\text{NOT}(R(x))$ negation of quantifiers 15

23 $\exists x \in D.\forall y \in D.(R(y)\ \text{IMPLIES}\ A(x,y))$

15 Suppose NOT($\exists x \in D.R(x)$)

16 $\forall x \in D.$NOT($R(x)$) negation of quantifiers 15

17 Let $x \in D$ be arbitrary.

22 $\forall y \in D.(R(y)$ IMPLIES $A(x, y))$

23 $\exists x \in D.\forall y \in D.(R(y)$ IMPLIES $A(x, y))$ construction 17, 22

15 Suppose NOT$(\exists x \in D.R(x))$

16 $\forall x \in D.$NOT$(R(x))$ negation of quantifiers 15

17 Let $x \in D$ be arbitrary.

18 Let $y \in D$ be arbitrary.

21 $R(y)$ IMPLIES $A(x, y)$

22 $\forall y \in D.(R(y)$ IMPLIES $A(x, y))$ generalization 18, 21

23 $\exists x \in D.\forall y \in D.(R(y)$ IMPLIES $A(x, y))$ construction 17,22

15 Suppose NOT($\exists x \in D.R(x)$)

16 $\forall x \in D.$NOT($R(x)$) negation of quantifiers 15

17 Let $x \in D$ be arbitrary.

18 Let $y \in D$ be arbitrary.

19 Assume NOT($A(x, y)$)

20 NOT($R(y)$)

21 $R(y)$ IMPLIES $A(x, y)$ indirect proof 19, 20

22 $\forall y \in D.(R(y)$ IMPLIES $A(x, y))$ generalization 18, 21

23 $\exists x \in D.\forall y \in D.(R(y)$ IMPLIES $A(x, y))$ construction 17,22

15 Suppose NOT($\exists x \in D.R(x)$)

16 $\forall x \in D.$NOT($R(x)$) negation of quantifiers 15

17 Let $x \in D$ be arbitrary.

18 Let $y \in D$ be arbitrary.

19 Assume NOT($A(x, y)$).

20 NOT($R(y)$) specialization 16

21 $R(y)$ IMPLIES $A(x, y)$ indirect proof 19, 20

22 $\forall y \in D.(R(y)$ IMPLIES $A(x, y))$ generalization 18, 21

23 $\exists x \in D.\forall y \in D.(R(y)$ IMPLIES $A(x, y))$ construction 17,22

1 Assume $\forall x \in D.(R(x)$ IFF $\forall y \in D.A(x,y))$.
2 P OR NOT(P) tautology
3 $(\exists x \in D.R(x))$ OR NOT$(\exists x \in D.R(x))$ substitution 2
4 Suppose $\exists x \in D.R(x)$
5 Let $d \in D$ be such that $R(d)$ instantiation 4
6 $R(d)$ IFF $\forall y \in D.A(d,y)$ specialization 1
7 $\forall y \in D.A(d,y)$ modus ponens 6,5
8 Let $y \in D$ be arbitrary.
9 Assume $R(y)$.
10 $A(d,y)$ specialization 7
11 $R(y)$ IMPLIES $A(d,y)$ direct proof 9,10
12 $\forall y \in D.(R(y)$ IMPLIES $A(d,y))$ generalization 8,11
13 $\exists x \in D.\forall y \in D.(R(y)$ IMPLIES $A(x,y))$ construction 5,12
14 $[\exists x \in D.R(x)]$ IMPLIES $[\exists x \in D.\forall y \in D.(R(y)$ IMPLIES $A(x,y))]$ direct proof 4,13
15 Suppose NOT$(\exists x \in D.R(x))$.
16 $\forall x \in D.$NOT$(R(x))$ negation of quantifiers 15
17 Let $x \in D$ be arbitrary.
18 Let $y \in D$ be arbitrary.
19 Assume NOT$(A(x,y))$.
20 NOT$(R(y))$ specialization 16
21 $R(y)$ IMPLIES $A(x,y)$ indirect proof 19,20
22 $\forall y \in D.(R(y)$ IMPLIES $A(x,y))$ generalization 18,21
23 $\exists x \in D.\forall y \in D.(R(y)$ IMPLIES $A(x,y))$ construction 17,22
24 [NOT$(\exists x \in D.R(x))$] IMPLIES $\exists x \in D.\forall y \in D.(R(y)$ IMPLIES $A(x,y))]$ direct proof 15,23
25 $\exists x \in D.\forall y \in D.(R(y)$ IMPLIES $A(x,y))$ proof by cases 3,14,24.
26 $[\forall x \in D.(R(x)$ IFF $\forall y \in D.A(x,y))]$ IMPLIES $[\exists x \in D.\forall y \in D.(R(y)$ IMPLIES $A(x,y))]$ direct proof 1,25

Problem 5

Give a well structured informal proof of:

Theorem $\forall n \in \mathbb{Z}^+ . ((\log_7 n \notin \mathbb{Q}) \text{ OR } (\log_7 n \in \mathbb{Z}))$.

Problem 5

Theorem $\forall n \in \mathbb{Z}^{+}.((\log_7 n \notin \mathbb{Q}) \text{ OR } (\log_7 n \in \mathbb{Z}))$.

Every real number is either irrational or rational.

$\log_7 n$ is rational implies that $\log_7 n$ is irrational or an integer.
$\log_7 n$ is irrational also implies that $\log_7 n$ is irrational or an integer.
Using proof by cases, it follows that $\log_7 n$ is irrational or an integer.

Problem 5

Theorem $\forall n \in \mathbb{Z}^+.((\log_7 n \notin \mathbb{Q}) \text{ OR } (\log_7 n \in \mathbb{Z}))$.

Every real number is either irrational or rational.

Suppose $\log_7 n$ is rational.

Hence $\log_7 n$ is an integer.

Therefore $\log_7 n$ is rational implies that $\log_7 n$ is irrational or an integer.
$\log_7 n$ is irrational also implies that $\log_7 n$ is irrational or an integer.
Using proof by cases, it follows that $\log_7 n$ is irrational or an integer.

Problem 5

Theorem $\forall n \in \mathbb{Z}^+.((\log_7 n \notin \mathbb{Q})$ OR $(\log_7 n \in \mathbb{Z}))$.

Every real number is either irrational or rational.

Suppose $\log_7 n$ is rational.

Then, by definition, there exist relatively prime non-negative integers p and $q \neq 0$ such that $\log_7 n = p/q$.

Multiplying both sides by q gives $\log_7 n^q = q \log_7 n = p$, so $n^q = 7^{\log_7 n^q} = 7^p$.

Since the factorization of a positive integer into primes is unique and 7 is prime, it follows that $n = 7^k$, where k is a non-negative integer.

Hence $\log_7 n$ is an integer.

Therefore $\log_7 n$ is rational implies that $\log_7 n$ is irrational or an integer.

$\log_7 n$ is irrational also implies that $\log_7 n$ is irrational or an integer.

Using proof by cases, it follows that $\log_7 n$ is irrational or an integer.

Theorem $\forall n \in \mathbb{Z}^+.((\log_7 n \notin \mathbb{Q}) \text{ OR } (\log_7 n \in \mathbb{Z}))$.

Let $n \in \mathbb{Z}^+$ be arbitrary.

$\log_7 n$ is irrational or an integer.

By generalization, the claim is true for all positive integers n.

Theorem $\forall n \in \mathbb{Z}^+.((\log_7 n \notin \mathbb{Q}) \text{ OR } (\log_7 n \in \mathbb{Z}))$.

Let $n \in \mathbb{Z}^+$ be arbitrary.

Therefore $\log_7 n$ is rational implies that $\log_7 n$ is an integer.
Since P IMPLIES Q is logically equivalent to NOT(P) OR Q,
$\log_7 n$ is irrational or an integer.

By generalization, the claim is true for all positive integers n.

Theorem $\forall n \in \mathbb{Z}^+.((\log_7 n \notin \mathbb{Q}) \text{ OR } (\log_7 n \in \mathbb{Z}))$.

Let $n \in \mathbb{Z}^+$ be arbitrary.

Suppose $\log_7 n$ is rational.

Hence $\log_7 n$ is an integer.

Therefore $\log_7 n$ is rational implies that $\log_7 n$ is an integer.

Since P IMPLIES Q is logically equivalent to NOT(P) OR Q,

$\log_7 n$ is irrational or an integer.

By generalization, the claim is true for all positive integers n.

Theorem $\forall n \in \mathbb{Z}^+.((\log_7 n \notin \mathbb{Q}) \text{ OR } (\log_7 n \in \mathbb{Z}))$.

Let $n \in \mathbb{Z}^+$ be arbitrary.

Suppose $\log_7 n$ is rational.

Then, by definition, there exist relatively prime non-negative integers p and $q \neq 0$ such that $\log_7 n = p/q$.

Multiplying both sides by q gives $\log_7 n^q = q \log_7 n = p$,

so $n^q = 7^{q \log_7 n} = 7^p$.

Since the factorization of a positive integer into primes is unique and 7 is prime, $n = 7^k$, where k is a non-negative integer.

Hence $\log_7 n$ is an integer.

Therefore $\log_7 n$ is rational implies that $\log_7 n$ is an integer.

Since P IMPLIES Q is logically equivalent to NOT(P) OR Q, $\log_7 n$ is irrational or an integer.

By generalization, the claim is true for all positive integers n.

Regular Tutorial 3 : Writing Formal Proofs

Logan Murphy

CSC 240

January 27th 2021

Contents

- Quiz Solution
- Proving Properties of Sets
- Problem Session

Quiz

Give a well structured, informal proof that, for all integers x and y, if xy is even, then x is even or y is even. Use proper indentation and state what proof techniques you are using.

Quiz Solution

Let x and y be integers.

Suppose that neither x nor y is even.

Then there exist integers u and v such that $x = 2u + 1$ and $y = 2v + 1$.

Since $xy = (2u + 1)(2v + 1) = 4uv + 2u + 2v + 1 = 2(2uv + u + v) + 1$, it follows that xy is not even.

Hence, by indirect proof, if xy is even then x is even or y is even.

By generalization, it follows that, for all integers x and y, if xy is even, then x is even or y is even.

Proving Properties of Sets: The Subset Relation

$A \subseteq B$ means

$$\forall x \in A. x \in B$$

or equivalently,

$$\forall x \in U.\ x \in A \ \text{IMPLIES}\ \ x \in B$$

.

Proving Properties of Sets: The Subset Relation

In either case, to prove the $A \subseteq B$:

Let $x \in A$ be arbitrary

$\vdots$

$x \in B$

$x \in A$ IMPLIES $x \in B$ direct proof

$\forall x \in A.\ x \in B$ generalization

Proving Properties of Sets: Set Equality

Two sets are equal when they contain exactly the same elements.

$$A = B$$

$$\forall x \in U.\ x \in A \text{ IFF } x \in B$$

$$\forall x \in U.(x \in A \text{ IMPLIES } x \in B) \text{ AND } (x \in B \text{ IMPLIES } x \in A)$$

$$A \subseteq B \text{ AND } B \subseteq A$$

Set equality is *mutual inclusion*.

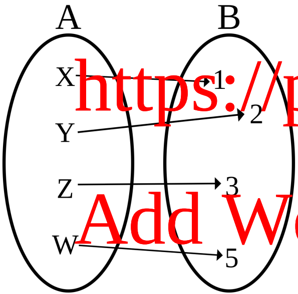
Proving Properties of Sets: Size Equality

The size (cardinality) of a set A, denoted $|A|$, is the number of elements in A.

So two sets are the same size if they contain the same number of elements.

Proving Properties of Sets: Size Equality

To prove $|A| = |B|$, prove there exists a bijective function from A to B.



A function $f : A \to B$ is bijective if and only if for every $b \in B$, there is exactly one $a \in A$ such that $f(a) = b$.

Also called a one-to-one correspondence.

Proving Properties of Sets: Size Equality

Example.

Let $A = \{a_0, a_1, ..., a_{m-1}\}$ be a set of size m.

Let $B = \{b_0, b_1, ..., b_{n-1}\}$ be a set of size n.

Prove that $|A \times B| = mn$ by by defining a bijection from $A \times B$ to the nonnegative integers $\{0, ..., mn - 1\}$.

Proving Properties of Sets: Size Equality

Solution.

$(A \neq \emptyset \ \text{AND} \ B \neq \emptyset) \ \text{OR} \ (A = \emptyset \ \text{OR} \ B = \emptyset)$ tautology

Assume $A = \emptyset \ \text{OR} \ B = \emptyset$.

Assume $A = \emptyset$. Then $m = 0$, so $mn = 0$.

If $A = \emptyset$, then $A \times B = \emptyset$, and thus $|A \times B| = 0$

$A = \emptyset \ \text{IMPLIES} \ |A \times B| = mn$ direct proof

$B = \emptyset \ \text{IMPLIES} \ |A \times B| = mn$ symmetry

$(A = \emptyset \ \text{OR} \ B = \emptyset) \ \text{IMPLIES} \ |A \times B| = mn$ proof by cases

Assume $A \neq \emptyset$ AND $B \neq \emptyset$

Let $f : A \times B \to \{0, ..., mn - 1\}$ be defined by

$$f(a_i, b_j) = (i \cdot n) + j$$

where $0 \leq i < m$ and $0 \leq j < n$

Given any integer k and $n \neq 0$, there exists a unique integer q (the quotient) and a unique integer r (the remainder) such that $0 \leq r < n$ and $k = q \cdot n + r$.

Since the quotient and remainder are unique, and we assumed $B \neq \emptyset$, this function is indeed a bijection.

Proving Properties of Sets: Size Equality

1.$(A \neq \emptyset \text{ AND } B \neq \emptyset) \text{ OR } (A = \emptyset \text{ OR } B = \emptyset)$ tautology

2. Assume $A = \emptyset \text{ OR } B = \emptyset$.

3. Assume $A = \emptyset$. Then $m = 0$, so $mn = 0$.

4. If $A = \emptyset$, then $A \times B = \emptyset$, and thus $|A \times B| = 0$

5. $A = \emptyset \text{ IMPLIES } |A \times B| = mn$ direct proof 2, 4

6. $B = \emptyset \text{ IMPLIES } |A \times B| = mn$ symmetry 4

7. $(A = \emptyset \text{ OR } B = \emptyset) \text{ IMPLIES } |A \times B| = mn$ proof by cases 5,6

8. Assume $A \neq \emptyset \text{ AND } B \neq \emptyset$

9. Let $f : A \times B \to \{0, \ldots, mn - 1\}$ be defined by $f(a_i, b_j) = (i \cdot n) + j$ where $0 \leq i < m$ and $0 \leq j < n$

10. Given any integer k and $n \neq 0$, there exists a unique integer q (the quotient) and unique integer r (the remainder) such that $0 \leq r < n$ and $k = q \cdot n + r$.

11. $B \neq \emptyset \text{ IMPLIES } f$ is a bijection direct proof 8, 10

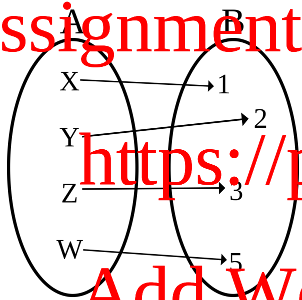
12. $A \neq \emptyset \text{ AND } B \neq \emptyset \text{ IMPLIES } |A \times B| = mn$ direct proof 8, 11

13. $(A \neq \emptyset \text{ AND } B \neq \emptyset) \text{ OR } (A = \emptyset \text{ OR } B = \emptyset) \text{ IMPLIES } |A \times B| = mn$ proof by cases 7, 12

14. $|A \times B| = mn$ modus ponens 1, 13

Proving Properties of Sets: Size Equality

Usually it takes more work to prove a function is bijective!



A function $f : A \to B$ is bijective if and only if for every $b \in B$, there is exactly one $a \in A$ such that $f(a) = b$.

I.e. f is bijective if and only if it is both injective and surjective.

Recall :

f is injective if and only if

$\forall a_1 \in A. \forall a_2 \in A. f(a_1) = f(a_2)$ IMPLIES $a_1 = a_2$

f is surjective if and only if $\forall b \in B. \exists a \in A. b = f(a)$

Proving Properties of Sets: Size Equality

To prove $|A| = |B|$, prove there exists a bijective function from A to B

1. Let $f : A \to B$ be the function defined by $f(a) = \_\_$

$\vdots$

4. f is injective

$\vdots$

8. f is surjective
9. f is bijective 4,8

10. $\exists f : A \to B$, such that f is bijective construction 1
11. Thus $|A| = |B|$ direct proof

Proving Properties of Sets: Size (In)equality

More generally, given sets A and B

- $|A| \leq |B|$ if and only if there exists an injective function from A to B

- $|A| \geq |B|$ if an only if there exists a surjective function from A to B

- $|A| = |B|$ if an only if there exists a bijective function from A to B

Problem 1

Give a well structured informal proof of:

THEOREM Let X be a set and let $A, B, C \subseteq X$.

Suppose that $A \cap B = A \cap C$ and $(X - A) \cap B = (X - A) \cap C$.

Then $B = C$.

You may use use the following lemma without proof.

LEMMA For all sets U and V, $U \cap V \subseteq U$ and $U \cap V \subseteq V$.

Assume $A, B, C \subseteq X$, $A \cap B = A \cap C$, and $(X - A) \cap B = (X - A) \cap C$.

Thus $B = C$.

Assume $A, B, C \subseteq X$, $A \cap B = A \cap C$, and $(X - A) \cap B = (X - A) \cap C$.

$B \subseteq C$.
By symmetry, $C \subseteq B$.
Thus $B = C$.

We can use symmetry, because the assumptions and statement do not change when all occurrences of B and C are interchanged.

Assume $A, B, C \subseteq X$, $A \cap B = A \cap C$, and $(X - A) \cap B = (X - A) \cap C$.

Let $b \in B$.

Then $b \in C$

Hence $B \subseteq C$.

By symmetry, $C \subseteq B$.

Thus $B = C$.

Assume $A, B, C \subseteq X$, $A \cap B = A \cap C$, and $(X - A) \cap B = (X - A) \cap C$.

Let $b \in B$.

Since $B \subseteq X$, it follows that $b \in X$.

Then $b \in C$

Hence $B \subseteq C$.

By symmetry, $C \subseteq B$.

Thus $B = C$.

Assume $A, B, C \subseteq X$, $A \cap B = A \cap C$, and $(X - A) \cap B = (X - A) \cap C$.

Let $b \in B$.

Since $B \subseteq X$, it follows that $b \in X$.

$b \in A$ or $b \in X - A$

Then $b \in C$

Hence $B \subseteq C$.

By symmetry, $C \subseteq B$.

Thus $B = C$.

Assume $A, B, C \subseteq X$, $A \cap B = A \cap C$, and $(X - A) \cap B = (X - A) \cap C$.

Let $b \in B$.

Since $B \subseteq X$, it follows that $b \in X$.

$b \in A$ or $b \in X - A$.

Therefore $b \in A$ IMPLIES $b \in C$.

Therefore $b \in (X - A)$ IMPLIES $b \in C$.

Then $b \in C$, using proof by cases.

Hence $B \subseteq C$.

By symmetry, $C \subseteq B$.

Thus $B = C$.

Assume $A, B, C \subseteq X$, $A \cap B = A \cap C$, and $(X - A) \cap B = (X - A) \cap C$.

Let $b \in B$.

Since $B \subseteq X$, it follows that $b \in X$.

$b \in A$ or $b \in X - A$.

Suppose $b \in A$.

Thus, $b \in C$.

Therefore $b \in A$ IMPLIES $b \in C$.

Suppose $b \in X - A$.

Thus, $b \in C$.

Therefore $b \in (X - A)$ IMPLIES $b \in C$.

Then $b \in C$, using proof by cases.

Hence $B \subseteq C$.

By symmetry, $C \subseteq B$.

Thus $B = C$.

Assume $A, B, C \subseteq X$, $A \cap B = A \cap C$, and $(X - A) \cap B = (X - A) \cap C$.

Let $b \in B$.

Suppose $b \in A$.

Thus, $b \in C$.

Assume $A, B, C \subseteq X$, $A \cap B = A \cap C$, and $(X - A) \cap B = (X - A) \cap C$.

Let $b \in B$.

Suppose $b \in A$.

Then $b \in A \cap B$.

Thus, $b \in C$.

Assume $A, B, C \subseteq X$, $A \cap B = A \cap C$, and $(X - A) \cap B = (X - A) \cap C$.

Let $b \in B$.

Suppose $b \in A$.

Then $b \in A \cap B$.

Since $A \cap B = A \cap C$,

Thus, $b \in C$.

Assume $A, B, C \subseteq X$, $A \cap B = A \cap C$, and $(X - A) \cap B = (X - A) \cap C$.

Let $b \in B$.

Suppose $b \in A$.

Then $b \in A \cap B$.

Since $A \cap B = A \cap C$,

it follows that $b \in A \cap C$.

Thus, $b \in C$.

Assume $A, B, C \subseteq X$, $A \cap B = A \cap C$, and $(X - A) \cap B = (X - A) \cap C$.

Let $b \in B$.

Suppose $b \in A$.

Then $b \in A \cap B$.

Since $A \cap B = A \cap C$,

it follows that $b \in A \cap C$.

By the lemma, $A \cap C \subseteq C$.

Thus, $b \in C$.

Assume $A, B, C \subseteq X$, $A \cap B = A \cap C$, and $(X - A) \cap B = (X - A) \cap C$.

Let $b \in B$.

Suppose $b \in X - A$.

Thus, $b \in C$.

Assume $A, B, C \subseteq X$, $A \cap B = A \cap C$, and $(X - A) \cap B = (X - A) \cap C$.

Let $b \in B$.

Suppose $b \in X - A$.

Then $b \in (X - A) \cap B$.

Thus, $b \in C$.

Assume $A, B, C \subseteq X$, $A \cap B = A \cap C$, and $(X - A) \cap B = (X - A) \cap C$.

Let $b \in B$.

Suppose $b \in X - A$.

Then $b \in (X - A) \cap B$.

Since $(X - A) \cap B = (X - A) \cap C$,

Thus, $b \in C$.

Assume $A, B, C \subseteq X$, $A \cap B = A \cap C$, and $(X - A) \cap B = (X - A) \cap C$.

Let $b \in B$.

Suppose $b \in X - A$.

Then $b \in (X - A) \cap B$.

Since $(X - A) \cap B = (X - A) \cap C$,

it follows that $b \in (X - A) \cap C$.

Thus, $b \in C$.

Assume $A, B, C \subseteq X$, $A \cap B = A \cap C$, and $(X - A) \cap B = (X - A) \cap C$.

Let $b \in B$.

Suppose $b \in X - A$.

Then $b \in (X - A) \cap B$.

Since $(X - A) \cap B = (X - A) \cap C$,

it follows that $b \in (X - A) \cap C$.

By the lemma, $(X - A) \cap C \subseteq C$.

Thus, $b \in C$.

Assume $A, B, C \subseteq X$, $A \cap B = A \cap C$, and $(X - A) \cap B = (X - A) \cap C$.

Let $b \in B$.

Since $B \subseteq X$, it follows that $b \in X$.

$b \in A$ or $b \in X - A$.

Suppose $b \in A$.

Then $b \in A \cap B$.

Since $A \cap B = A \cap C$, it follows that $b \in A \cap C$.

By the lemma, $A \cap C \subseteq C$.

Thus, $b \in C$.

Therefore $b \in A$ IMPLIES $b \in C$.

Suppose $b \in X - A$.

Then $b \in (X - A) \cap B$.

Since $(X - A) \cap B = (X - A) \cap C$, it follows that $b \in (X - A) \cap C$.

By the lemma, $(X - A) \cap C \subseteq C$.

Thus, $b \in C$.

Therefore $b \in (X - A)$ IMPLIES $b \in C$.

Then $b \in C$ using proof by cases.

Hence $B \subseteq C$.

By symmetry, $C \subseteq B$.

Thus $B = C$.

Problem 2

(a) Indicate what is wrong with this solution and give suggestions for improvement, as if you are a TA for CSC240.

First, we show that $B \subseteq C$.

Let $p \in A \cap B$. This means that $p \in A$ and $p \in B$.
Since $A \cap B = A \cap C$, $A \cap B \subseteq A \cap C$ and $A \cap C \subseteq A \cap B$.
That means that $p \in A \cap B$ implies $p \in A \cap C$.
Since $p \in A \cap C$, it follows that $p \in C$.
Because $p \in A \cap B$ means that $p \in B$, $p \in B$ implies $p \in C$.
Therefore $B \subseteq C$.

Second, we show that $C \subseteq B$.
Let $p \in (X - A) \cap C$.
This means that $p \in C$.
Since $(X - A) \cap C = (X - A) \cap B$, we know that $(X - A) \cap C \subseteq (X - A) \cap B$.
This means that $p \in (X - A) \cap C$ implies $p \in (X - A) \cap B$.
Since $p \in (X - A) \cap B$, it can be said that $p \in B$.
Since $p \in (X - A) \cap C$ means $p \in C$, $p \in C$ implies $p \in B$.
Therefore $C \subseteq B$.

We have shown both that $B \subseteq C$ and $C \subseteq B$. Therefore, $B = C$.

Problem 2

Some problems

- Use of the word "mean" is ambigous (sometimes used in place of IFF, other times used as IMPLIES)

- Proof of subset relation is improper; should begin with $p \in B$.
- What about the case if $p \notin A$?
- From $p \in A \cap B$ IMPLIES $p \in B$ and $p \in A \cap B$ IMPLIES $p \in C$, we cannot infer $B \subseteq C$

(b) Indicate what is wrong with this solution and give suggestions for improvement, as if you are a TA for CSC240.

First, I will show that $B \subseteq C$.
Let $x \in B$. Considering $A \cap B = A \cap C$, $x \in A \cap B$ implies that $x \in A \cap C$.
From the lemma, $A \cap B \subseteq B$ and $A \cap C \subseteq C$.
Since $A \cap B = A \cap C$, it can be said that $A \cap B \subseteq C$.
Since $x \in A \cap B$ and therefore $x \in B$, and $A \cap B \subseteq C$, x is therefore also an element of C in the case that $A \cap B = A \cap C$.

Likewise, $(X - A) \cap B = (X - A) \cap C$ implies that a given element x exists in $(X - A) \cap B$ and $(X - A) \cap C$.
Therefore $x \notin A$ and $x \in B$. Also, according to the lemma, $(X - A) \cap C \subseteq C$.
Since $(X - A) \cap C = (X - A) \cap B$, then $(X - A) \cap B \subseteq C$.
This implies that there is an element $x \in (X - A) \cap B$ and $x \in C$.
The phrase $x \in (X - A) \cap B$ can be further broken down to $x \notin A$ and $x \in B$.
Therefore $B \subseteq C$ regardless of whether $x \in A$ or $x \notin A$.

To prove that $C \subseteq B$, it suffices to show that element $y \in C$ implies $y \in B$.
Take $(X - A) \cap B = (X - A) \cap C$.
Then $y \in (X - A) \cap B$, and thus $y \in B$ and $y \notin A$.
According to the lemma, $(X - A) \cap B \subseteq B$.
Since $(X - A) \cap B = (X - A) \cap C$, then $(X - A) \cap C \subseteq B$.
This implies that $y \notin A$ and $y \in C$. Also$y \in B$.
Therefore, $C \subseteq B$. If $B \subseteq C$ and $C \subseteq B$, then $B = C$.

Problem 2

Some problems

- Proof by cases is incorrect. The proper case analysis would be on $(x \in A \cap B)\ \mathrm{OR}\ (x \in (X - A) \cap B)$, not $x \in A\ \mathrm{OR}\ x \notin A$.

- From the equality in line 6, you can't infer an existential; what if all the sets are empty?
- Paragraph 3 contains redundant information! E.g. $y \in B$ is deduced twice.

(c) Indicate what is wrong with this solution and give suggestions for improvement, as if you are a TA for CSC240.

Let $x \in A \cap B$.
Then $x \in A$ and $x \in B$. Also let $x \in A \cap C$, then $x \in A$ and $x \in C$.
This shows that $x \in A$, B and C.

Since $(X - A) \cap B = (X - A) \cap C$, then $x \in X$ and $x \notin A$ and $x \in B$ and $x \in X$ and $x \notin A$ and $x \in C$ are equivalent.
Therefore $x \in X$, B, and C and $x \notin A$.
By the lemma, $A \cap B \subseteq A$ and $A \cap B \subseteq B$, and $A \cap C \subseteq A$ and $A \cap C \subseteq C$.
However, since $x \notin A$, then $B = C$.

Problem 2

Some problems

- x is used for two different "arbitrary" variables; each assumption needs a fresh name for the variables!

- Line 5 does not follow from line 4.
- Conclusion is not justified from previous sentences.

Solutions to

CSC240 Winter 2021 Homework Assignment 3

1. (a)

1 Let $i \in \mathbb{Z}^+$ be arbitrary.
2 (i mod $2 = 0$) OR (i mod $4 = 1$) OR (i mod $4 = 3$), property of $\mathbb{Z}^+$.
3 Suppose i mod $2 = 0$.
4 $element(S', i, 2)$, by definition of S', 3.
5 Let $j \in \mathbb{Z}^+$ be arbitrary.
6 Let $k = 2(\lceil j/2 \rceil + 1) \in \mathbb{Z}^+$.
7 $2\lceil j/2 \rceil \geq j$, property of $\mathbb{Z}^+$.
8 $2(\lceil j/2 \rceil + 1) = 2\lceil j/2 \rceil + 2 > j$, property of $\mathbb{Z}^+$, 7.
9 $k > j$, substitution 6, 8.
10 $lt(j, k)$, by definition of lt, 9.
11 k mod $2 = 0$, property of $\mathbb{Z}^+$, 6
12 $element(S', k, 2)$, by definition of S'.
13 $lt(j, k)$ AND $element(S', k, 2)$, conjunction: 10,12.
14 $\exists k \in \mathbb{Z}^+.(lt(j, k)$ AND $element(S', k, 2))$, direct proof of existential quantification: 6,13.
15 $\forall j \in \mathbb{Z}^+.\exists k \in \mathbb{Z}^+.(lt(j, k)$ AND $element(S', k, 2))$, generalization: 5,14.
16 $element(S', i, 2)$ AND $\forall j \in \mathbb{Z}^+.\exists k \in \mathbb{Z}^+.(lt(j, k)$ AND $element(S', k, 2))$, conjunction: 4,15.

17 $\exists x \in \mathbb{Z}.[element(S', i, x)$ AND $\forall j \in \mathbb{Z}^+.\exists k \in \mathbb{Z}^+.(lt(j, k)$ AND $element(S', k, x))]$, direct proof of existential quantification: 16, since $2 \in \mathbb{Z}$.
18 (i mod $2 = 0$) IMPLIES $\exists x \in \mathbb{Z}.[element(S', i, x)$ AND $\forall j \in \mathbb{Z}^+.\exists k \in \mathbb{Z}^+.(lt(j, k)$ AND $element(S', k, x))]$, direct proof of implication: 3,17.

19 Suppose i mod $4 = 1$.
20 $element(S', i, 1)$, by definition of S', 19.
21 Let $j \in \mathbb{Z}^+$ be arbitrary.
22 Let $k = 4\lceil j/2 \rceil + 1 \in \mathbb{Z}^+$.

23 $4\lceil j/2 \rceil \geq j$, property of $\mathbb{Z}^+$.
24 $4\lceil j/2 \rceil + 1 > j$, property of $\mathbb{Z}^+$, 23.
25 $k > j$, substitution 22, 24.
26 $lt(j, k)$, by definition of lt, 25.
27 k mod $4 = 1$, property of $\mathbb{Z}^+$, 22.
28 $element(S', k, 1)$, by definition of S', 27.
29 $lt(j, k)$ AND $element(S', k, 1)$, conjunction: 26,28.
30 $\exists k \in \mathbb{Z}^+.(lt(j, k)$ AND $element(S', k, 1))$, direct proof of existential quantification: 22,29.
31 $\forall j \in \mathbb{Z}^+.\exists k \in \mathbb{Z}^+.(lt(j, k)$ AND $element(S', k, 1))$, generalization: 21,30.
32 $element(S', i, 1)$ AND $\forall j \in \mathbb{Z}^+.\exists k \in \mathbb{Z}^+.(lt(j, k)$ AND $element(S', k, 1))$, conjunction: 20,31.
33 $\exists x \in \mathbb{Z}.[element(S', i, x)$ AND $\forall j \in \mathbb{Z}^+.\exists k \in \mathbb{Z}^+.(lt(j, k)$ AND $element(S', k, x))]$, direct proof of existential quantification: 32, since $1 \in \mathbb{Z}$.
34 (i mod $4 = 1$) IMPLIES $\exists x \in \mathbb{Z}.[element(S', i, x)$ AND $\forall j \in \mathbb{Z}^+.\exists k \in \mathbb{Z}^+.(lt(j, k)$ AND $element(S', k, x))]$, direct proof of implication: 19,33.

35 Suppose $i \bmod 4 = 3$.

36 $element(S', i, 3)$, by definition of S', 35.

37 Let $j \in \mathbb{Z}^+$ be arbitrary.

38 Let $k = 4\lceil j/2 \rceil + 3 \in \mathbb{Z}^+$.

39 $4\lceil j/2 \rceil \geq j$, property of $\mathbb{Z}^+$.

40 $4\lceil j/2 \rceil + 1 > j$, property of $\mathbb{Z}^+$, 39.

41 $k > j$, substitution 38, 40.

42 $lt(j, k)$, by definition of lt, 41.

43 $k \bmod 4 = 3$, property of $\mathbb{Z}^+$, 38.

44 $element(S', k, 3)$,by definition of S', 43.

45 $lt(j, k)$ AND $element(S', k, 3)$, conjunction: 42,44.

46 $\exists k \in \mathbb{Z}^+.(lt(j, k)$ AND $element(S', k, 3))$, direct proof of existential quantification: 38,45.

47 $\forall j \in \mathbb{Z}^+.\exists k \in \mathbb{Z}^+.(lt(j, k)$ AND $element(S', k, 3))$, generalization: 37, 47.

48 $element(S', i, 3)$ AND $\forall j \in \mathbb{Z}^+.\exists k \in \mathbb{Z}^+.(lt(j, k)$ AND $element(S', k, 3))$, conjunction: 36,47

49 $\exists x \in \mathbb{Z}.[element(S', i, x)$ AND $\forall j \in \mathbb{Z}^+.\exists k \in \mathbb{Z}^+.(lt(j, k)$ AND $element(S', k, x))]$, direct proof of existential quantification: 49, since $3 \in \mathbb{Z}$.

50 $(i \bmod 4 = 3)$ IMPLIES $\exists x \in \mathbb{Z}.[element(S', i, x)$ AND $\forall j \in \mathbb{Z}^+.\exists k \in \mathbb{Z}^+, .(lt(j, k)$ AND $element(S', k, x))]$, direct proof of implication: 35,49.

51 $\exists x \in \mathbb{Z}.[element(S', i, x)$ AND $\forall j \in \mathbb{Z}^+.\exists k \in \mathbb{Z}^+.(lt(j, k)$ AND $element(S', k, x))]$, proof by cases: 2,18,34,50

52 $\forall i \in \mathbb{Z}^+.\exists x \in \mathbb{Z}.[element(S', i, x)$ AND $\forall j \in \mathbb{Z}^+.\exists k \in \mathbb{Z}^+.(lt(j, k)$ AND $element(S', k, x))]$, generalization: 1,51.

53 $io(S')$, by defintion of io, 52

(b)

1 Let $i = 4 \in \mathbb{Z}^+$.

2 Let $x \in \mathbb{Z}$ be arbitrary.

3 Suppose $element(S'', i, x)$.

4 $x = 1$, by definition of S'' 1, 3.

5 Let $j = i \in \mathbb{Z}^+$.

6 Let $k \in \mathbb{Z}^+$ be arbitrary.

7 Suppose $lt(j, k)$.

8 NOT($element(S'', k, x)$), definition of S'' 1, 5, 7.

9 $lt(j, k)$ IMPLIES NOT($element(S'', k, x)$), direct proof of implication 7, 8.

10 (A IMPLIES NOT(B)) IMPLIES NOT(A AND B), tautology

11 [$lt(j, k)$ IMPLIES NOT($element(S'', k, x)$)] IMPLIES [NOT($lt(j, k)$ AND $element(S'', k, x)$)], substitution 10.

12 NOT($lt(j, k)$ AND $element(S'', k, x)$), modus ponens 11, 9.

13 $\forall k \in \mathbb{Z}^+$.NOT($lt(j, k)$ AND $element(S'', k, x)$), generalization 6, 12.

14 NOT[$\exists k \in \mathbb{Z}^+$.($lt(j, k)$ AND $element(S'', k, x)$)], negation of quantifier 13 .

15 $\exists j \in \mathbb{Z}^+$.NOT[$\exists k \in \mathbb{Z}^+$.($lt(j, k)$ AND $element(S'', k, x)$)], construction 5, 14

16 NOT[$\forall j \in \mathbb{Z}^+.\exists k \in \mathbb{Z}^+$.($lt(j, k)$ AND $element(S'', k, x)$)], negation of quantifier 15.

17 $element(S'', i, x)$ IMPLIES NOT[$\forall j \in \mathbb{Z}^+.\exists k \in \mathbb{Z}^+$.($lt(j, k)$ AND $element(S'', k, x)$)], direct proof of implication 3, 16.

18 ($element(S'', i, x)$ IMPLIES NOT[$\forall j \in \mathbb{Z}^+.\exists k \in \mathbb{Z}^+$.($lt(j, k)$ AND $element(S'', k, x)$)]) IMPLIES NOT[($element(S'', i, x)$ AND $\forall j \in \mathbb{Z}^+.\exists k \in \mathbb{Z}^+$.($lt(j, k)$ AND $element(S'', k, x)$)], substitution 10

19 NOT[($element(S'', i, x)$ AND $\forall j \in \mathbb{Z}^+.\exists k \in \mathbb{Z}^+$.($lt(j, k)$ AND $element(S'', k, x)$)], modus ponens 18, 17

20 $\forall x \in \mathbb{Z}$.NOT[$element(S'', i, x)$ AND $\forall j \in \mathbb{Z}^+.\exists k \in \mathbb{Z}^+$.($lt(j, k)$ AND $element(S'', k, x)$)] generalization 2, 19.

21 NOT[$\exists x \in \mathbb{Z}$.[($element(S'', i, x)$ AND $\forall j \in \mathbb{Z}^+.\exists k \in \mathbb{Z}^+$.($lt(j, k)$ AND $element(S'', k, x)$)], negation of quantifier 20.

22 $\exists i \in \mathbb{Z}^+$.NOT[$\exists x \in \mathbb{Z}$.[($element(S'', i, x)$ AND $\forall j \in \mathbb{Z}^+.\exists k \in \mathbb{Z}^+$.($lt(j, k)$ AND $element(S'', k, x)$)], construction 1, 21

23 NOT($\forall i \in \mathbb{Z}^+.\exists x \in \mathbb{Z}$.[($element(S'', i, x)$ AND $\forall j \in \mathbb{Z}^+.\exists k \in \mathbb{Z}^+$.($lt(j, k)$ AND $element(S'', k, x)$)]), negation of quantifier 22.

24 NOT($io(S'')$), by defintion of io, 23

CSC 240 Lecture 4   Induction

Let p: ℕ → {T,F} be a predicate.
Suppose you want to prove ∀ n ∈ ℕ. p(n)

⋮
p(0)   basis or base case

⋮

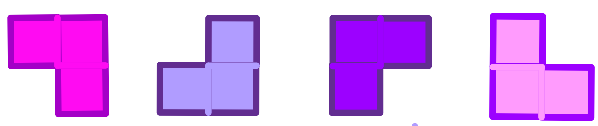
∀ n ∈ ℕ.[p(n) IMPLIES p(n+1)]
∀ n ∈ ℕ. p(n) by induction

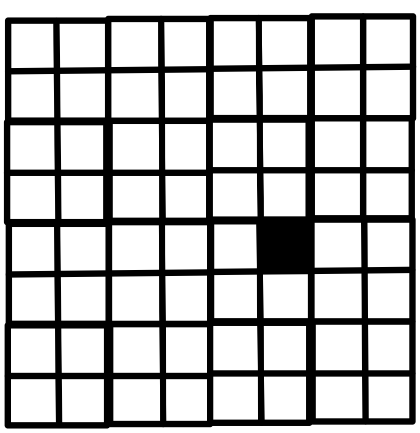
To prove ∀ n ∈ ℕ.[p(n) IMPLIES p(n+1)]:
  Let n ∈ ℕ be arbitrary
    Assume p(n)
    ⋮
    p(n+1)
  p(n) IMPLIES p(n+1) direct proof
∀ n ∈ ℕ.[p(n) IMPLIES p(n+1)] generalization

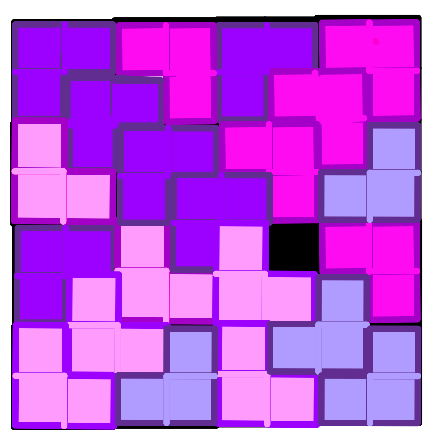
THEOREM 1. Consider any square chessboard whose sides have length which is a power of 2. If any one square is removed, then the resulting shape can be tiled using only 3-square L-shaped tiles.

A shape is tiled by a set of tiles means that you can fit the tiles together with no gaps or overlaps to fill the shape exactly. Another name for a tiling is a tessellation.

3-square L-shaped tiles
Let's call these L-tiles







Proof:
Let P(n) = "any 2^n x 2^n chessboard with one square removed can be tiled using 3-square L-shaped tiles".

Let C(n) denote the set of all 2^n x 2^n chessboards with one square removed.

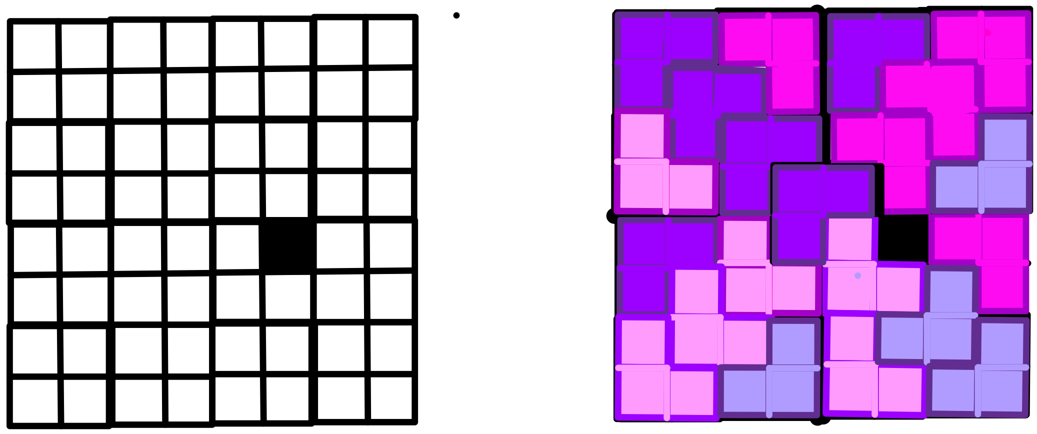
P(n) = "$\forall$ c $\in$ C(n). c can be tiled using ~~only~~ L-tiles"

Prove $\forall n \in \mathbb{N}$. P(n)

Basis:
P(0) is true, since a 2^0 x 2^0 chessboard with one square removed has no squares and hence can be tiled with 0 L-tiles.

Let n $\in \mathbb{N}$ be arbitrary.
Suppose that P(n) is true.
Let c in C(n+1) be arbitrary.
Divide c into 4 equal 2^n x 2^n chessboards
One of these has a square removed, so it is in C(n).
By the induction hypothesis, it can be tiled using L-tiles.
Consider the other three 2^n x 2^n chessboards.
Each has 1 square that is in the centre of c.
With that square removed, the induction hypothesis says that it can be tiled with L-tiles.
The 3 squares that we removed from the centre of c can be tiled using one L-tile.
Hence c can be tiled using L -tiles.
$\forall$ c $\in$ C(n+1). c can be tiled using only L-tiles. by generalization
P(n+1) by definition of P(n+1)
P(n) IMPLIES P(n+1) by direct proof
$\forall$ n $\in \mathbb{N}$. (P(n) IMPLIES P(n+1)) by generalization
$\forall$ n $\in \mathbb{N}$. P(n) by induction

Example for n = 2



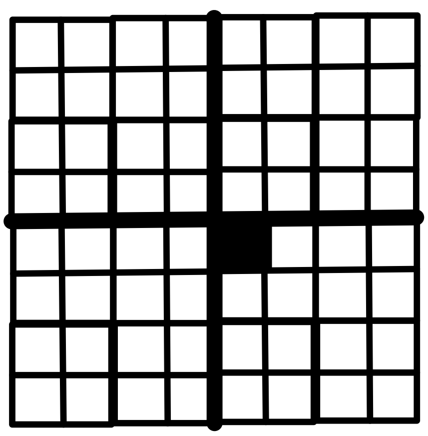
THEOREM 1. All square chessboards with sides whose length is a power of 2 and has one square removed can be tiled using only L-tiles.

THEOREM 2. All square chessboards with sides whose length is a power of 2 and has one square removed from the middle can be tiled using only L-tiles.

Let C'(n) denote the set of all 2^n x 2^n chessboards with one square removed from the middle

Let P'(n) = "$\forall$ c $\in$ C'(n). c can be tiled using only L-tiles"

How can P'(n) be use to prove P'(n+1)? It can't!



Proving a harder/more general result is easier here because strengthening the induction hypothesis makes the proof of the induction step easier.

THEOREM 3 For all $n \in \mathbb{N}$. $2n + 1 \leq 2^n$.

For all $n \in \mathbb{N}$,
let $q(n)$ = " $2n + 1 \leq 2^n$ "

Base case:
$2 \times 0 + 1 = 1 \leq 1 = 2^0$
so $q(0)$ is true.

Let $n \in \mathbb{N}$ be arbitrary.
Assume $q(n)$
$\vdots$
$q(n+1)$

DOESN'T WORK:
true for $n = 0$
false for $n = 1$: $2 \times 1 + 1 = 3 > 2 = 2^1$
false for $n = 2$: $2 \times 2 + 1 = 5 > 4 = 2^2$
true for $n \geq 3$.

The actual theorem is
THEOREM 3 $\forall n \in \mathbb{N}$. ($n \geq 3$ IMPLIES $q(n)$)
or, alternatively,
$\forall m \in M$. $q(m)$,
where $M = \{m \in \mathbb{N} \mid m \geq 3\}$.

How can this be proved by induction?

IDEA 1

Let $r(n)$ = "($n \geq 3$) IMPLIES $q(n)$".
Then $\forall n \in \mathbb{N}$. $r(n)$ means the same as $\forall n \in \mathbb{N}$. (($n \geq 3$) IMPLIES $q(n)$)

base case: $r(0)$
vacuously true, since $0 \geq 3$ is false.

induction step:

Let $n \in \mathbb{N}$ be arbitrary.

Assume $r(n)$

$\vdots$

$r(n+1)$

$r(n)$ IMPLIES $r(n+1)$

$\forall n \in \mathbb{N}.\ (r(n)$ IMPLIES $r(n+1))$ by generalization

$\forall n \in \mathbb{N}.\ r(n)$ by induction

Assume $(n \geq 3)$ IMPLIES $q(n)$

$\vdots$

$(n+1 \geq 3)$ IMPLIES $q(n+1)$

Assume $(n \geq 3)$ IMPLIES $q(n)$

Assume $n+1 \geq 3$

Then either $n = 2$ or $n \geq 3$.

Case 1: $n = 2$

Then $2(n+1) + 1 = 2 \times 3 + 1 = 7 \leq 2^3 = 2^{n+1}$, so $q(n+1)$ is true

Case 2: $n \geq 3$

$q(n)$, modus ponens

$2n + 1 \leq 2^n$, by definition of $q(n)$

$2 \leq 8 = 2^3 \leq 2^n$ arithmetic

$2(n+1) + 1 = 2n + 2 + 1 = (2n+1) + 2 \leq 2^n + 2^n = 2^{n+1}$

substitution

$q(n+1)$, by definition of $q(n+1)$

$q(n+1)$ proof by cases

$(n+1 \geq 3)$ IMPLIES $q(n+1)$, direct proof of implication

IDEA 2

Let $p(n) = q(n+3)$.

Then $\forall\, n \in \mathbb{N}.\ p(n)$ means the same as $\forall\, m \in M.\ q(m)$

and

$\forall\, n \in \mathbb{N}.\ (p(n)$ IMPLIES $p(n+1))$ means the same as

$\forall\, m \in M.\ (q(m)$ IMPLIES $q(m+1))$.

base case: $p(0)$

induction step:

Let $n \in \mathbb{N}$ be arbitrary.

Assume $p(n)$

⋮

$p(n+1)$

$p(n)$ IMPLIES $p(n+1)$

$\forall\, n \in \mathbb{N}.\ (p(n)$ IMPLIES $p(n+1))$

by generalization

$\forall\, n \in \mathbb{N}.\ p(n)$

by induction

base case: $q(3)$

induction step:

Let $m \in M$ be arbitrary

Assume $q(m)$

⋮

$q(m+1)$

$q(m)$ IMPLIES $q(m+1)$

$\forall\, m \in M.\ (q(m)$ IMPLIES $q(m+1))$

by generalization

$\forall\, m \in M.\ q(m)$

by induction

to prove

$\forall\, n \in \mathbb{N}.\ ((n \geq b)$ IMPLIES $p(n))$

it suffices to prove

$p(b)$

and either

$\forall\, n \in \{m \in \mathbb{N} \mid m \geq b\}.\ (p(n)$ IMPLIES $p(n+1))$

or

$\forall\, n \in \mathbb{N}.\ (((n \geq b)$ AND $p(n))$ IMPLIES $p(n+1))$

Suppose we want to prove that a predicate q is true only for all even natural numbers?

Let $p(k) = q(2k)$

Then $\forall\, k \in \mathbb{N}.\ p(k)$

means the same as

$\forall\, k \in \mathbb{N}.\ q(2k)$

which means the same as $\forall\, n \in \mathbb{N}.\ (n$ is even IMPLIES $q(n))$.

Base case:
p(0) = q(0)

induction step is: p(k) IMPLIES p(k+1)
which is q(2k) IMPLIES q(2k+2)

Therefore, it is sufficient to prove
q(0) and
$\forall$ n $\in \mathbb{N}$. (q(n) IMPLIES q(n+2))

This isn't necessary:
we're proving
q(0) IMPLIES q(2)
q(1) IMPLIES q(3)
q(2) IMPLIES q(4)
q(3) IMPLIES q(5)

We don't need half of these. In fact, they might not even be true.
For example

$n^2 \leq 2^n$ is true for all even $n \geq 0$ and for $n = 1$
but is false for $n = 3$
Thus q(1) IMPLIES q(3) is false.

To prove $\forall$ i $\in$ {0,1,...,n}. p(i)

Base case:
⋮
p(0)

Induction step:
  Let $\boxed{i \in \{0,1,\ldots,n-1\}}$ be arbitrary.
    Assume p(i).
    ⋮
    p(i+1)
  p(i) IMPLIES p(i+1), direct proof
$\forall$ i $\in$ {0,1,...,n-1}. (p(i) IMPLIES p(i+1)) generalization
$\forall$ i $\in$ {0,1,...,n}. p(i) induction

arithmetic mean $(a_1 + ... + a\_n)/n = (\Sigma\{a_i \mid 1 \le i \le n\}) / n$
geometric mean $(a_1 ... a\_n)$ ^{1/n} = $(\Pi\{a_i \mid 1 \le i \le n\})$ ^{1/n}

THEOREM 4 For all positive integers n and all positive real numbers a_1 ,..., a_n their geometric mean is less than or equal to their arithmetic mean.

For all n ∈ ℤ+, let P(n) =
"∀ a ∈ (ℝ+)^{1,...,n}. [(Π{a(i) | 1 ≤ i ≤ n}) ^{1/n} ≤ (Σ{a(i) | 1 ≤ i ≤ n}) / n]"

THEOREM 4 ∀ n ∈ ℤ+. P(n)

Base case: n = 2
⋮
Induction steps:
  Let n be an arbitrary integer ≥ 2.
    Assume P(n)
    ⋮
    P(n-1)
  P(n) IMPLIES P(n-1)
    Assume P(n)
    ⋮
    P(2n)
  P(n) IMPLIES P(2n)
∀ n ∈ ℤ+. P(n) by induction

Why?
To prove P(m), let k be the smallest power of 2 greater than or equal to m
2^{k-1} < m ≤ 2^k

Using the base case, prove P(2)
Using the second induction step,
prove P(4), P(8),..., P(2^k)
Using the first induction step,
prove P(2^k - 1), P(2^k - 2),...,P(m)

Base case: n = 2

Let $a_1, a_2 \in \mathbb{R}^+$ be arbitrary

$$a_1^2 - 2a_1a_2 + a_2^2 = (a_1 - a_2)^2 \geq 0$$

$$a_1^2 + a_2^2 \geq 2a_1a_2$$

$$\left(\frac{a_1 + a_2}{2}\right)^2 = \frac{a_1^2 + a_2^2 + 2a_1a_2}{4}$$

$$\geq \frac{2a_1a_2 + 2a_1a_2}{4} = a_1a_2$$

$$\frac{a_1 + a_2}{2} \geq (a_1a_2)^{1/2}$$

By generalization $P(2)$ is true.

Let n be an arbitrary integer ≥ 2.

Assume P(n)

Let $a_1, \ldots, a_{n-1} \in \mathbb{R}^+$ be arbitrary

Let $b_i = a_i$ for $i = 1, \ldots, n-1$ and

let $b_n = (a_1 + \cdots + a_{n-1})/(n-1)$

Then, by specialization of P(n)

$$\prod_{i=1}^{n} b_i \leq \left[\left(\sum_{i=1}^{n} b_i\right)/n\right]^n = \left[\left(\sum_{i=1}^{n-1} a_i + b_n\right)/n\right]^n$$

$$= \left[((n-1)b_n + b_n)/n\right]^n = \left(\frac{nb_n}{n}\right)^n = b_n^n$$

Thus $\prod_{i=1}^{n-1} a_i = \prod_{i=1}^{n-1} b_i = \left(\prod_{i=1}^{n} b_i\right)/b_n \leq \frac{b_n^n}{b_n}$

$$= b_n^{n-1} = \left(\left(\sum_{i=1}^{n-1} a_i\right)/(n-1)\right)^{n-1}$$

P(n-1) by generalization.

P(n) IMPLIES P(n-1)

Assume P(n)

Let $a_1, \dots, a_{2n} \in \mathbb{R}^+$ be arbitrary

Let $b_1 = \frac{1}{n}\sum_{i=1}^{n} a_i$ and $b_2 = \frac{1}{n}\sum_{i=n+1}^{2n} a_i$

By specialization of P(n)

$$\prod_{i=1}^{n} a_i \le \left[\left(\sum_{i=1}^{n} a_i\right)/n\right]^n$$ and

$$\prod_{i=n+1}^{2n} a_i \le \left[\left(\sum_{i=n+1}^{2n} a_i\right)/n\right]^n$$

and by specialization of P(2)

$$b_1 b_2 \le \left[(b_1 + b_2)/2\right]^2$$

Hence $$\prod_{i=1}^{2n} a_i \le \left(\left(\sum_{i=1}^{n} a_i\right)/n\right)^n \left[\left(\sum_{i=n+1}^{2n} a_i\right)/n\right]^n$$

$$= (b_1 b_2)^n \le \left(\frac{b_1 + b_2}{2}\right)^{2n} = \left(\frac{\frac{1}{n}\sum_{i=1}^{2n} a_i}{2}\right)^{2n} = \left(\frac{\sum_{i=1}^{2n} a_i}{2n}\right)^{2n}$$

P(2n) by generalization

P(n) IMPLIES P(2n)

STRONG or COMPLETE INDUCTION

to prove
$\forall\, i \in \mathbb{N}.\ p(i)$
it suffices to prove
$\forall\, i \in \mathbb{N}.\ ([\ \forall\, j \in \mathbb{N}.\ ((j < i) \text{ IMPLIES } p(j))\ ] \text{ IMPLIES } p(i)\ )$

Let $i \in \mathbb{N}$ be arbitrary
Assume $\forall\, j \in \mathbb{N}.\ (\ (j < i) \text{ IMPLIES } p(j\ ))$
$\vdots$
various cases
$\vdots$
$p(i)$
$[\ \forall\, j \in \mathbb{N}.\ (\ (j < i) \text{ IMPLIES } p(j))\ ] \text{ IMPLIES } p(i)$ direct proof
$\forall\, i \in \mathbb{N}.\ ([\ \forall\, j \in \mathbb{N}.\ ((j < i) \text{ IMPLIES } p(j))\ ] \text{ IMPLIES } p(i)\ )$ generalization
THESE 2 LINES ARE OPTIONAL
$\forall\, i \in \mathbb{N}.\ p(i)$ strong induction

Alternative:

Base Case:
$\vdots$
$p(0)$

Induction Step:
Let $i \in \mathbb{Z}+$ be arbitrary.
Assume $\forall\, j \in \mathbb{N}.\ (\ (j < i) \text{ IMPLIES } p(j)\ )$
$\vdots$
$p(i)$
$\forall\, i \in \mathbb{N}.\ p(i)$ strong induction

Theorem For all $n \geq 4$, exactly \$n can be made using only toonies and \$5 bills.

Proof: For all $n \in \mathbb{N}$,
let P(n) = "exactly \$n can be made using only toonies and \$5 bills." or,
let $P(n)$ = "$\exists f \in \mathbb{N}. \exists g \in \mathbb{N}. (n = 2f + 5g)$"

Let $n \in \mathbb{N}$ be arbitrary.
Suppose $n \geq 4$ and
$\forall j \in \mathbb{N}. ((4 \leq j < n) \text{ IMPLIES } P(j))$

if $n = 4$, then $n = 2 \times 2 + 0 \times 5$
if $n = 5$, then $n = 0 \times 2 + 1 \times 5$
if $n \geq 6$, then
  $4 \leq n-2 < n$
  $P(n-2)$ specialization
  so $\exists f \in \mathbb{N}. \exists g \in \mathbb{N}. (n-2 = 2f + 5g)$
  thus $n = 2(f+1) + 5g$
$P(n)$ proof by cases
So $n \geq 4$ IMPLIES $P(n)$ direct proof
$\forall n \in \mathbb{N}. (n \geq 4 \text{ IMPLIES } P(n))$ strong induction

Complete induction from Simple induction

"strong" "weak"

Weak: If we know $P(0)$ and

$$\forall n \in \mathbb{N} \left(P(n) \text{ IMPLIES } P(n+1) \right)$$

Then we can infer $\forall n \in \mathbb{N}.\ P(n)$

Strong: $\forall i \in \mathbb{N} \left(\left[\forall j \in \mathbb{N}. \left(j < i \text{ IMPLIES } Q(j) \right) \right] \text{ IMPLIES } Q(i) \right)$

Then we can infer $\forall i \in \mathbb{N}.\ Q(n)$

$$R(i) = \forall j \in \mathbb{N}. \left(j < i \text{ IMPLIES } Q(j) \right)$$

Strong: $\boxed{\forall i \in \mathbb{N} \left(R(i) \text{ IMPLIES } Q(i) \right)}$ ☆

Assume that we have proved the strong induction hypothesis

Base case: $R(0)$ says $\forall j \in \mathbb{N}. \left(j < 0 \text{ IMPLIES } Q(j) \right)$

Since $j < 0$ is false for all $j \in \mathbb{N}$

$(j < 0 \text{ IMPLIES } Q(j))$ is vacuously true.

Induction step:

Let $n \in \mathbb{N}$ be arbitrary

Induction hypothesis: Assume $R(n)$ $\forall j \in \mathbb{N} (j < n \text{ IMPLIES } Q(j))$

We wish to prove $R(n+1)$ $\forall j \in \mathbb{N} (j < n+1 \text{ IMPLIES } Q(j))$

Let $j \in \mathbb{N}$ be arbitrary

Case 0: $j > n+1$. vacuously true

Case 1: $j < n$. Then true by induction hypothesis: $Q(j)$ is true

Case 2: $j = n$

By strong induction hypothesis (☆)

$R(n)$ IMPLIES $Q(n)$

We assume $R(n)$ by induction hypothesis

And so $Q(n)$

In other words, $R(n+1)$ is true.

<u>Claim</u> By simple/weak induction, $\forall n \in \mathbb{N}$, $R(n)$

For all $n \geq 0$, $2 + 3 + 4 + \ldots + n = n(n+1)/2$

Proof. For n in $\mathbb{N}$, let $P(n) =$ "$2 + 3 + 4 + \cdots + n = n(n+1)/2$".

Base case: $P(0)$ is true, since both sides of the equation are equal to zero. (Recall that a sum with no terms is zero.)

Inductive step: Now we must show that $P(n)$ implies $P(n+1)$ for all $n \geq 0$. Let n in $\mathbb{N}$ be arbitrary and suppose that $P(n)$ is true; that is, $2+3+4+\cdots+n = n(n+1)/2$. Then we can reason as follows: $2+3+4+\cdots+n+(n+1) = (2+3+4+\cdots+n)+(n+1) = n(n+1)/2+(n+1) = (n+1)(n+2)/2$ Above, we group some terms, use the assumption $P(n)$, and then simplify. This shows that $P(n)$ implies $P(n+1)$. By the principle of induction, $P(n)$ is true for all $n \geq 0$.

Where exactly is the error in this proof?

$n = 0$

$$\underbrace{2 + 3 + \ldots + (n+1)}_{\text{no terms in this sum}}$$

$$=$$

$$(2 + 3 + \ldots + n) + \underbrace{(n+1)}_{\text{this sum has one term!}}$$

For n in N, let $P(n)$ = "postage of exactly n cents can be made using only 4-cent and 6-cent stamps".

Consider the following complete induction "proof" of the statement "$P(n)$ holds for all $n \geq 4$".

Base case: $n = 4$. Postage of exactly 4 cents can be made using just a single 4-cent stamp. So, $P(4)$ holds.

Induction Step: Let $i \geq 4$ be an arbitrary integer, and suppose that $P(j)$ holds for all j such that $4 \leq j < i$.That is, for all j such that $4 \leq j < i$, postage of exactly j cents can be made using only 4-cent and 6-cent stamps. We must prove that $P(i)$ holds. That is, we must prove that postage of exactly i cents can be made using only 4-cent and 6-cent stamps.

Since $\boxed{i-4} < i$, by induction hypothesis we can make postage of exactly $i-4$ cents using only 4-cent and 6-cent stamps. Suppose this requires k 4-cent stamps and m 6-cent stamps; that is, $i - 4 = 4k + 6m$.

Let $k' = k + 1$ and $m' = m$. We have

$$\begin{aligned} 4k' + 6m' &= 4(k+1) + 6m, && \text{by definition of } k' \text{ and } m' \\ &= 4k + 6m + 4 = (i-4) + 4 && \text{by the induction hypothesis.} \end{aligned}$$

Thus, $P(i)$ holds.

We can't make an odd amount of postage using only 4-cent and 6-cent stamps. Thus, the statement "$P(n)$ holds for all $n \geq 4$" is certainly false. Consequently, the above "proof" is incorrect. What is wrong with it?

$i = 4$ is not the only base case

4, 5, 6, 7

For any integer $n \geq 2$, let $A(n)$ denote the number of ways to fully parenthesize a sum of n terms such as $a_1 + \cdots + a_n$.
For example, $A(2) = 1$, since the only way to fully parenthesize $a_1 + a_2$ is $(a_1 + a_2)$ and $A(3) = 2$, since the only ways to fully parenthesize $a_1+a_2+a_3$ are $((a_1+a_2)+a_3)$ and $(a_1+(a_2+a_3))$.

Prove by induction that $A(n) \leq (n-1)^{n-1}$ for all integers $n \geq 2$.

For all integers $n \geq 2$, let $P(n)$ denote the predicate "$A(n) \leq (n-1)^{n-1}$".
Prove that $\forall n \in \{n \in \mathbb{N} \mid n \geq 2\}.P(n)$.

\# of ways to parenthesize 2 terms

$a+b = (a+b)$

$((a+b)+c)$ $(a+(b+c))$

$((a)+(b+c))$

For any integer $n \geq 2$, let $A(n)$ denote the number of ways to fully parenthesize a sum of n terms such as $a_1 + \cdots + a_n$.
For example, $A(2) = 1$, since the only way to fully parenthesize $a_1 + a_2$ is $(a_1 + a_2)$ and $A(3) = 2$, since the only ways to fully parenthesize $a_1+a_2+a_3$ are $((a_1+a_2)+a_3)$ and $(a_1+(a_2+a_3))$.

Prove by induction that $A(n) \leq (n-1)^{n-1}$ for all integers $n \geq 2$.

For all integers $n \geq 2$, let $P(n)$ denote the predicate "$A(n) \leq (n-1)^{n-1}$".
Prove that $\forall n \in \{n \in \mathbb{N} \mid n \geq 2\}.P(n)$.

Solution:
Note that $A(2) = 1 = 1^1$ and $A(3) = 2 \leq 4 = 2^2$. Let $n \geq 4$ and assume that $A(i) \leq (i-1)^{i-1}$ for $2 \leq i < n$.

Each way to fully parenthesize $a_1 + \cdots + a_n$ has one of the following three forms, where X_i denotes a fully parenthesized sum of $i \geq 2$ terms.

- $(a_1 + X_{n-1})$,
- $(X_i + X_{n-i})$ for $2 \leq i \leq n-2$, and
- $(X_{n-1} + a_n)$.

The number of expressions of each of the first and last forms is $A(n-1)$ and,
for $2 \leq i \leq n-2$, the number of expressions of the form $(X_i + X_{n-i})$ is $A(i) \cdot A(n-i)$.
Thus $A(n) = 2A(n-1) + \sum_{i=2}^{n-2} A(i)A(n-i)$.

By the induction hypothesis, $A(i) \leq (i-1)^{i-1}$ for $2 \leq i < n$, so

$$\begin{aligned}
A(n) &\leq 2(n-2)^{n-2} + \sum_{i=2}^{n-2} (i-1)^{i-1}(n-i-1)^{n-i-1} \\
&\leq 2(n-1)^{n-2} + \sum_{i=2}^{n-2} (n-1)^{i-1}(n-1)^{n-i-1} \\
&= 2(n-1)^{n-2} + \sum_{i=2}^{n-2} (n-1)^{n-2} \\
&= 2(n-1)^{n-2} + (n-3)(n-1)^{n-2} \\
&= (2+n-3)(n-1)^{n-2} = (n-1)^{n-1}
\end{aligned}$$

Hence the claim is true for n. By induction, $A(n) \leq (n-1)^{n-1}$ for all integers $n \geq 2$.

1. Assume $\forall i \in \mathbb{N}. ([\forall n \in \mathbb{N}. (j < i \text{ IMPLIES } Q(j))] \text{ IMPLIES } Q(i))$.

2. $\forall i \in \mathbb{N}. (R(i) \text{ IMPLIES } Q(i))$ definition of R, 1

3. Let $j \in \mathbb{N}$ be arbitrary.

4. NOT$(j < 0)$ property of $\mathbb{N}$

5. NOT(A) IMPLIES $(A$ IMPLIES $B)$ tautology

6. NOT$(j < 0)$ IMPLIES $(j < 0$ IMPLIES $Q(i))$ substitution 5

7. $j < 0$ IMPLIES $Q(j)$ modus ponens 4, 6

8. $\forall j \in \mathbb{N}. (j < 0 \text{ IMPLIES } Q(j))$ generalization 3, 7

9. $R(0)$ definition of R, 8

10. Let $n \in \mathbb{N}$ be arbitrary.

11. Assume $R(n)$.

12. $\forall j \in \mathbb{N}. (j < n \text{ IMPLIES } Q(j))$ definition of R, 11

13. Let $k \in \mathbb{N}$ be arbitrary.

14. Assume $k < n + 1$.

15. $k < n$ OR $k = n$ property of $\mathbb{N}$, 13, 14

16. $k < n$ IMPLIES $Q(k)$ specialization 12

17. Assume $k = n$.

18. $R(n)$ IMPLIES $Q(n)$ specialization 2

19. $Q(n)$ modus ponens 11, 18

20. $Q(k)$ substitution 17, 19

21. $k = n$ IMPLIES $Q(k)$ direct proof 17, 20

22. $Q(k)$ proof by cases 15, 16, 21

23. $(k < n + 1)$ IMPLIES $Q(k)$ direct proof 14, 22

24. $\forall k \in \mathbb{N}. ((k < n + 1) \text{ IMPLIES } Q(k))$ generalization 13, 23

25. $R(n + 1)$ definition of R, 24

26. $R(n)$ IMPLIES $R(n + 1)$ direct proof 11, 25

27. $\forall n \in \mathbb{N}. (R(n) \text{ IMPLIES } R(n + 1))$ generalization 10, 26

28. $\forall n \in \mathbb{N}. R(n)$ (simple) induction 9, 27

29. Let $i \in \mathbb{N}$ be arbitrary.

30. $R(i)$ specialization 28

31. $R(i)$ IMPLIES $Q(i)$ specialization 2

32. $Q(i)$ modus ponens 30, 31

33. $\forall i \in \mathbb{N}. Q(i)$ generalization 29, 32

34. $[\forall i \in \mathbb{N}. ([\forall n \in \mathbb{N}. (j < i \text{ IMPLIES } Q(j))] \text{ IMPLIES } Q(i))]$
IMPLIES $(\forall i \in \mathbb{N}. Q(i))$ direct proof 1, 33

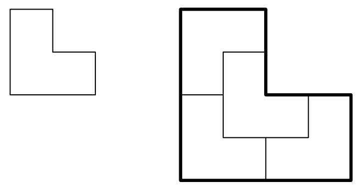
Solutions to

CSC240 Winter 2021 Homework Assignment 4

due Tuesday February 9, 2021

1. Prove by induction that, for all $n \geq 1$, an L-shaped region with two sides of length $2n$ and four sides of length n can be tiled using a sufficient number of L-tiles (i.e. L-shaped tiles each with two sides of length 2 and four sides of length 1).

   For example, when $n = 1$, the region can be tiled with one L-tile, and when $n = 2$, it can be tiled with four L-tiles, as follows:

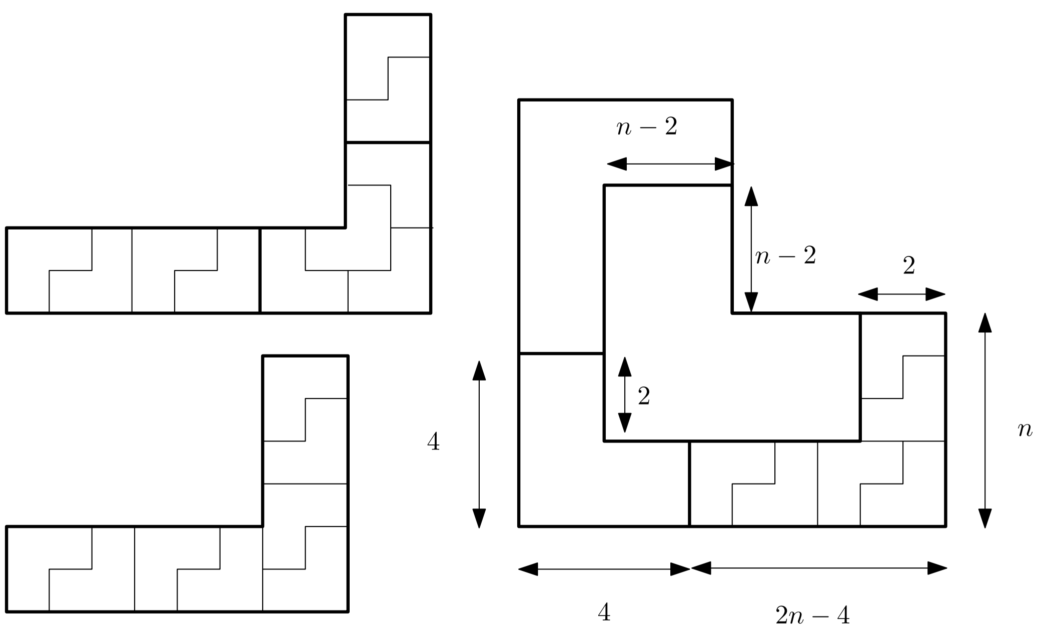


Solution: Let $R(n)$ denote an L-shaped region with two sides of length $2n$ and four sides of length n.
Let $P(n)$ denote the predicate "$R(n)$ can be tiled using L-tiles".

Let $n \in \mathbb{Z}^+$ be arbitrary and assume that $P(i)$ is true for $1 \leq i < n$.

$P(1)$ and $P(2)$ are true by the examples given in the question. So, suppose $n \geq 3$.

Note that $R(n)$ can be partitioned into an instance of $R(n-2)$, an instance of $R(2)$ and two instances of an L-shaped region with one side of length $2n-4$, one side of length $2n-6$, one side of length n, one side of length $n-2$, and two sides of length 2, as illustrated here.



Since $1 \le n-2 < n$ and $1 \le 2 < n$, it follows by the induction hypothesis that $R(n-2)$ and $R(2)$ can be tiled using L-times So, it remains to tile an L-shaped region with one side of length $2n-4$, one side of length $2n-6$, one side of length n, one side of length $n-2$, and two sides of length 2

If $n-2$ is divisible by 3, then this region can be partitioned into an $(n-2) \times 2$ rectangle and a $2 \times (2n-4)$ rectangle. Since a 2×3 rectangle can be tiled (with two tiles), and both $n-2$ and $2n-4 = 2(n-2)$ are divisible by 3, an $(n-2) \times 2$ rectangle and a $2 \times (2n-4)$ rectangle can both be tiled.

If n is divisible by 3, then this region can be partitioned into an $n \times 2$ rectangle and a $2 \times (2n-6)$ rectangle. Since both n and $2n-6$ are divisible by 3, an $n \times 2$ rectangle and a $2 \times (2n-6)$ rectangle can both be tiled.

If $n-1$ is divisible by 3, then this region can be partitioned into an instance of $R(2)$, an $(n-4) \times 2$ rectangle, and a $2 \times (2n-8)$ rectangle. Since both $n-4 = (n-1)-3$ and $2n-8 = 2(n-4)$ are divisible by 3, an $(n-4) \times 2$ rectangle and a $2 \times (2n-8)$ rectangle can both be tiled.

Thus, in all cases, $R(n)$ can be tiled. Hence $P(n)$ is true. By strong induction, $P(n)$ is true for all $n \ge 1$.

2. Consider a sequence in $\{H, T, B\}^n$. A *step* consists of selecting an occurrence of an H, changing it to a B and then flipping each of its (at most 2) neighbours from H to T or from T to H. If a neighbour is a B, it does not change. Prove by induction that, for all $n \ge 1$ and for all sequences $S \in \{H, T\}^n$, there is a sequence of steps that converts S to B^n if and only if S contains an odd number of H's.

Solution: Let $q : \mathbb{Z}^+ \to \{\mathrm{T}, \mathrm{F}\}$ be the predicate such that $q(n) =$ "$\forall S \in \{H, T\}^n$. [(there exists a sequence of steps that converts S to B^n) IFF (S contains an odd number of H's)]".

Let $n \in \mathbb{Z}^+$ be arbitrary and assume $q(i)$ is true for all i such that $1 \le i < n$.

Consider the case $n = 1$. Then $S = H$ or $S = T$. If $S = H$, then S contains exactly one H, and it is converted to B by applying a step to this H. If $S = T$, then S contains 0 occurrences of H, so no steps can be applied to it. Since 1 is odd and 0 is even, (there exists a sequence of steps that converts S to B^n) IFF (S contains an odd number of H's).

Now, consider the case that $n > 1$. First suppose that S contains an odd number of H's. Then $S = S'HS''$, where $S' = T^k$ for some $0 \le k < n$ and S'' is the suffix of S following its first occurrence of H. Note that S'' contains an even number of H's. Apply a step σ to the first occurrence of H in S. Let $S_1 = S_1'BS_1''$ be the resulting string, where $|S'| = |S_1'| = k$ and $|S''| = |S_1''| = n-1-k$.

If $k = 0$, then S' is an empty sequence of B's and $S_1' = S'$. In this case, let σ' be the empty sequence of steps. If $k > 0$, then $S_1' = T^{k-1}H$. By the induction hypothesis, there exists a sequence of steps σ' that converts S_1' to B^k. Since there is a B following S_1' in S_1, none of these steps affect S_1''.

If $n-k-1 = 0$, then S'' is an empty sequence of B's and $S_1'' = S''$, so $\sigma\sigma'$ converts S to B^n. Otherwise, $S_1'' \in \{H, T\}^{n-k-1}$. It contains an odd number of H's, since step σ changed the first letter of S'' from T to H or from H to T. By the induction hypothesis, there exists a sequence of steps σ'' that converts S_1'' to B^{n-k-1}, since $n-1-k < n$. Then $\sigma\sigma'\sigma''$ is a sequence of steps that converts S to B^n, because when σ'' is applied, the rest of the string consists of B's, which are not affected by any steps.

Hence, if S contains an odd number of H's, there exists a sequence of steps that converts S to B^n.

Now suppose there is a sequence of steps τ that converts S to B^n. Let $0 \leq i \leq n-1$ be the number of H's and T's that precede the H to which the first step of τ is applied, so $S = S'HS''$, where $|S'| = k < n$ and $|S''| = n-1-k < n$. Let $S_1 = S_1'BS_1''$ be the resulting string, where $|S'| = |S_1'|$ and $|S''| = |S_1''|$.

If $k \geq 1$, then the number of H's in S_1' differs from the number of H's in S' by 1. Likewise, if $n-1-k \geq 1$, the number of H's in S_1'' differs from the number of H's in S'' by 1. Let τ' be the subsequence of steps in τ that are applied to positions $1 \leq i \leq k$ and let τ'' be the subsequence of steps in τ that are applied to positions $k+1 \leq i \leq n$. Since there is a B in position k of S_1, the steps in τ' do not affect S_1'' and the steps in τ'' do not affect S_1'. Thus τ' converts S_1' to B^{k-1} and τ'' converts S_1'' to B^{n-1-k}.

If $k-1 \geq 1$, then, by the induction hypothesis, S_1' contains an odd number of H's and, hence S' contains an even number of H's. If $k-1 = 0$, then $S' = S_1'$ contains zero H's which is an even number.

Likewise, if $n-1-k \geq 1$ then, by the induction hypothesis, S_1'' contains an odd number of H's and, hence, S'' contains an even number of H's, and if $n-1-k = 0$, then $S'' = S_1''$ contains zero H's which is an even number. Since $S = S'HS''$, it follows that S contains an odd number of H's.

Hence, if there exists a sequence of steps that converts S to B^n, then S contains an odd number of H's,

Therefore, (there exists a sequence of steps that converts S to B^n) IFF (S contains an odd number of H's. By generalization, $q(n)$ is true. Finally, by strong induction, $\forall n \in \mathbb{Z}^+ . q(n)$ is true.

STRONG INDUCTION

Let $i \in \mathbb{N}$ be arbitrary
Assume $\forall j \in \mathbb{N}.$ (($j < i$) IMPLIES $p(j)$)
⋮
various cases
⋮
$p(i)$
optional: [$\forall j \in \mathbb{N}.$ (($j < i$) IMPLIES $p(j)$)] IMPLIES $p(i)$ direct proof
optional: $\forall i \in \mathbb{N}.$ ([$\forall j \in \mathbb{N}.$ (($j < i$) IMPLIES $p(j)$)] IMPLIES $p(i)$) generalization
$\forall i \in \mathbb{N}.\ p(i)$ strong induction

THEOREM Every integer greater than 1 is a product of primes.

Proof:

Let $M = \{ n \in \mathbb{N} \mid n > 1\}$.
For all $n \in M$, let $p(n)$ = "n is a product of primes"

Let $n \in M$ be arbitrary.
Suppose $\forall i \in M.$ (($i < n$) IMPLIES $p(i)$)

If n is prime, then n is a product of 1 prime.
Otherwise, by definition of prime,
$\exists$ k in M. $\exists$ m in M. ($n = mk$)
Since $k < n$ and $m < n$, it follows by the induction hypothesis, that
k is a product of primes and m is a product of primes.
Thus $n = km$ is a product of primes, i.e. $p(n)$.

By strong induction, $\forall n \in M.\ p(n)$

Proofs by weak induction can be easily transformed into proofs by strong induction by using the strong induction template instead of the ordinary induction template.
When proving $P(n+1)$, there is no harm in assuming $P(0),\ldots,P(n-1)$ in addition to $P(n)$.

Proofs by strong induction can also be transformed into proofs by weak induction. This was discussed in tutorial.

Recursively Defined Sets

Such definitions have 2 parts:
a base case, that doesn't depend on anything else, and
a constructor case, that depends on previous cases.

Examples

1. {0,1}* = set of all finite strings of bits

Base case:
the empty string, λ, of length 0, is in {0,1}*
Constructor case:
if s ∈ {0,1}* then s0 and s1 are in {0,1}*

More generally, if **Σ** is any finite set of letters,

then **Σ*** is the set of all finite strings of letter from **Σ**.
For example **Σ** = {[,]}

2. Brkts = set of finite strings of matched brackets

Base case:
the empty string, λ, of length 0, is in Brkts
Constructor cases:
if s,t ∈ Brkts then [s] ∈ Brkts and st ∈ Brkts
Alternative constructor case:
if s,t ∈ Brkts then [s]t ∈ Brkts

3. S = syntactically correct formulas of propositional logic

Base case:
propositional variables are in S
constructor cases:
If f, f' ∈ S, then
NOT(f) ∈ S
(f OR f') ∈ S
(f AND f') ∈ S
(f IMPLIES f') ∈ S
(f IFF f') ∈ S

(f XOR f') $\in$ S

4. M = syntactically correct monotone formulas of propositional logic
Base case:
propositional variables are in M
Constructor cases:
If f, f' $\in$ M, then
(f OR f') $\in$ M
(f AND f') $\in$ M

M is the smallest set of strings containing all the propositional variables that is closed under AND and OR.

- There can be many sets that satisfy a definition, for example S also satisfies the definition for M
- We define a set recursively, we mean (but may not say explicitly) that we are defining the smallest such set.

5. A = arithmetic expressions involving natural numbers
Base case: $\mathbb{N} \subseteq$ A
Constructor cases:
if e,f $\in$ A then
e + f $\in$ A and
e x f $\in$ A

When a recursive definition allows an element to be constructed in more than one way, the definition is ambiguous.
For example, 3 x 2 + 1
can be obtained by combining 3 x 2 and 1 with the first constructor case
or by combining 3 and 2 + 1 with the second constructor case

Notice:
- We are defining a set of finite strings
- In the base case(s), we explicitly define the smallest or simplest elements in the set.
- In the constructor case(s), we define the ways in which larger or more complex elements in the set can be constructed out of smaller or simpler elements

Structural induction

a form of induction used to prove properties about recursively defined sets.

Let S be a recursively defined set.
To prove $\forall x \in S.\ p(x)$, where $p{:}S \to \{T,F\}$ is a predicate,
prove:
p(x) for all base cases of the definition
p(x) for the constructor cases of the definition,
assuming it is true for the components of x

Example:

Let M be the set of syntactically correct monotone formulas of propositional logic.
For all $f \in M$,
let Nv(f) = "the number of occurrences of propositional variables in f",
let Nc(f) = "the number of occurrences of connectives in f", and
let p(f) = "Nv(f) = 1 + Nc(f)".

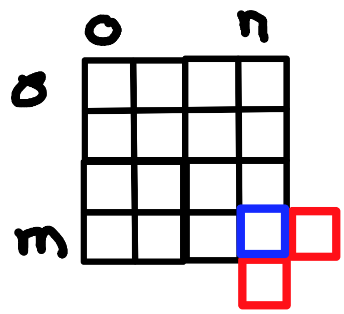
Base case:
If f is a propositional variable, then Nv(f) =1 and Nc(f) = 0, so p(f) is true.

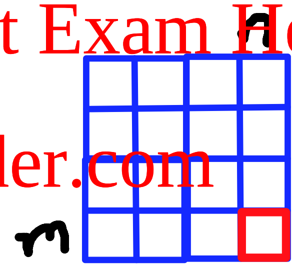
Constructor cases:
Let f' and f" in M be arbitrary.
Assume p(f') and p(f") are true.
Consider f = (f' OR f").
Nv(f) = Nv(f') + Nv(f")
= (1 + Nc(f')) +(1 + Nc(f")) by the induction hypothesis
= 1 + (Nc(f') + Nc(f") + 1)
= 1 + Nc(f).
Hence p(f) is true.
Similarly, if f = (f' AND f"), ~~then~~
~~Nv(f) = Nv(f') + Nv(f")~~
~~= (1 + Nc(f')) +(1 + Nc(f")) by the induction hypothesis~~
~~= 1 + (Nc(f') + Nc(f") + 1)~~
~~= 1 + Nc(f).~~
~~so p(f) is true.~~
By structural induction, $\forall f \in M.\ p(f)$.

Proving universally quantified predicates of two or more variables:
$\forall m \in \mathbb{N}. \forall n \in \mathbb{N}.\ p(m,n)$

$\mathbb{N} \times \mathbb{N}$ can also be defined recursively:
Base case: $(0,0) \in \mathbb{N} \times \mathbb{N}$
Constructor case: if $(m,n) \in \mathbb{N} \times \mathbb{N}$, then
$(m,n+1) \in \mathbb{N} \times \mathbb{N}$ and $(m+1,n) \in \mathbb{N} \times \mathbb{N}$.

To prove $\forall m \in \mathbb{N}. \forall n \in \mathbb{N}.\ p(m,n)$ using structural induction:
Base case:
Prove $p(0,0)$
Constructor case:
Let $(m,n) \in \mathbb{N} \times \mathbb{N}$ be arbitrary.
Assume $p(m,n)$.
Prove $p(m,n+1)$ and $p(m+1,n)$





Strong induction alternative:
Let $(m,n) \in \mathbb{N} \times \mathbb{N}$ be arbitrary.
Assume

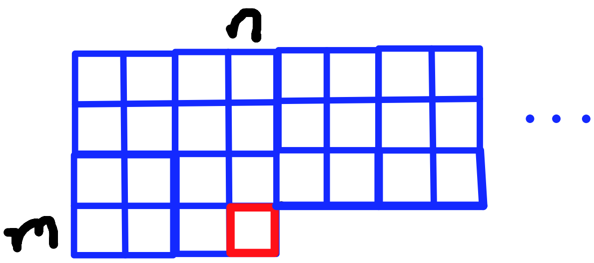
$\forall i \in \mathbb{N}. \forall j \in \mathbb{N}. ([(i \leq m) \text{ AND } (j \leq n) \text{ AND } ((i<m) \text{ OR } (j<n))] \text{ IMPLIES } p(i,j))$.
Prove $p(m,n)$

Another approach: define $Q(m) =$ "$\forall n \in \mathbb{N}.\ p(m,n)$"
To prove $\forall m \in \mathbb{N}. \forall n \in \mathbb{N}.\ p(m,n)$
it suffices to prove $\forall m \in \mathbb{N}.\ Q(m)$ by induction.

Let $m \in \mathbb{N}$ be arbitrary.
Assume $\forall i \in \mathbb{N}.\ (i < m \text{ IMPLIES } Q(i))$
Let $n \in \mathbb{N}$. be arbitrary
Assume $\forall j \in \mathbb{N}.\ (j < n \text{ IMPLIES } p(m,j))$
⋮
$p(m,n)$
$\forall j \in \mathbb{N}.\ p(m,j)$ strong induction
$Q(m)$ definition of Q
$\forall m \in \mathbb{N}.\ Q(m)$ by strong induction

To prove Q(m) we can use induction on n.

While proving p(m,n), we have assumed p(i,j)
for all (i,j) ∈ ℕ x ℕ such that (i < m) or (i = m and j < n)



Defining functions on recursively defined sets

Examples:

1. Let M be the set of syntactically correct monotone formulas of propositional logic.

Nv: M → ℕ

For f ∈ M, let
Nv(f) = "the number of occurrences of propositional variables in f"

Nv(X) = 1 for any propositional variable X
Nv(f) = Nv(f') + Nv(f") for f = (f' OR f") and for f = (f' AND f")

2. Let A be the set of arithmetic expressions involving natural numbers and addition and multiplication

value: A → ℕ
For f ∈ A, let
value(f) = "the natural number represented by f"

Base Case:
value(a) = a for all a ∈ ℕ

Constructor Cases: for all f,g $\in$ A
value(f+g) = value(f) + value(g)
value(fxg) = value(f) x value(g)

Problem: the function value is not well defined:
value(3 x 2 + 1) = value(3 x 2) + value(1) = 6 + 1 = 7
since value(3 x 2) = value(3) x value(2) = 3 x 2 = 6.

value(3 x 2 + 1) = value(3) x value(2 + 1) = 3 x 3 = 9
since value(2 + 1) = value(2) + value(1) = 2 + 1 = 3.

Be careful when defining functions on recursively defined sets that are ambiguous.

3. Let B denote the set of all binary trees.

Base case: the empty tree is in B

Constructor case: if t_1, $t_2 \in$ B, and r is a node, then


```
  r
 / \   ∈ B
t1  t2
```


t_1 and t_2 are called the left and right subtrees of t, respectively.
they are denoted left(t), right(t)

N:B $\rightarrow \mathbb{N}$
Base Case:
N(empty tree) = 0
Constructor case:
N(t) = 1 + N(left(t)) + N(right(t))
Then N(t) = number of nodes in t

L:B $\rightarrow \mathbb{N}$
Base Case:
L(empty tree) = 0
Constructor case:
L(t) = L(left(t)) + L(right(t))
Then L(t) = number of leaves in t

Actually, $L(t) = 0$ for all $t \in B$.
Must define a second base case,
L(one node tree) = 1, where a one node tree consists of a root with empty left and right subtrees.

Theorem
A binary tree with n nodes has at most $\lceil n/2 \rceil$ leaves.

For $q: B \to \{T,F\}$
Let $q(t) =$ " $(L(t) \leq \lceil N(t)/2 \rceil)$ "

We want to prove $\forall\ t \in B.q(t)$

For all $t \in B$ and $n \in \mathbb{N}$,
let $S(t,n) =$ "t has n nodes" and
let $AL(t,n) =$ "t has at most n leaves".
Let $p(n) =$ "$\forall\ t \in B.\ [S(t,n)$ IMPLIES $AL(t, \lceil n/2 \rceil)]$"

We want to prove $\forall\ n \in \mathbb{N}.\ p(n)$

Proof:
Let $t \in B$ be arbitrary.

Base Case 1: t is the empty tree
$N(t)=0$.
$L(t) = 0$,
so $L(t) = 0 = \lceil 0/2 \rceil = \lceil N(t)/2 \rceil$.

Hence $q(t)$.

Base Case 2: t is a one node tree
$N(t) = 1$ and $L(t) = 1$,
so $L(t) = 1 = \lceil 1/2 \rceil = \lceil N(t)/2 \rceil$.

Hence $q(t)$.

Constructor Case: t has at least 2 nodes.
$N(t) \geq 2$.
Assume $q(left(t))$ and $q(right(t))$.
Then $L(left(t)) \leq \lceil N(left(t))/2 \rceil$ and $L(right(t)) \leq \lceil N(right(t))/2 \rceil$.

Since
$L(t) = L(left(t)) + L(right(t))$ and
$N(t) = N(left(t)) + N(right(t) + 1$,
it follows that
$L(t) \leq \lceil N(left(t))/2 \rceil + \lceil N(right(t))/2 \rceil$

$\leq (N(left(t))+1)/2 + (N(right(t))+1)/2$
$= [N(left(t)) + N(right(t)) + 1 + 1]/2$
$= [N(t)+1]/2.$

We know $\lceil N(t)/2 \rceil \leq [N(t)+1]/2$, which is not useful.

Since L(t) is an integer, $L(t) \leq \lfloor [N(t)+1]/2 \rfloor = \lceil N(t)/2 \rceil$.

Hence q(t).

By structural induction, $\forall\ t \in B.q(t)$
$\forall\ t \in B.\ (L(t) \leq \lceil N(t)/2 \rceil)$ by definition

Structural induction can be proved using strong induction:

Let E_0 be the elements in the set S because of the Base Cases.
Let E_1 be the elements in the set S obtained from the elements of E_0 by applying the constructor case once.

Let E_i be the elements in the set S obtained from the elements of E_0 by applying the constructor case i times.

$q: \mathbb{N} \rightarrow \{T,F\}$
Let $q(i) =$ "$\forall\ t \in E_i.\ p(t)$"
Then $\forall\ i \in \mathbb{N}.\ q(i)$ means the same as $\forall\ t \in S.\ p(t)$
since $S = \cup\{ E_i \mid i \in \mathbb{N} \}$.

Sometimes a proof by induction can be disguised as a proof by contradiction.
The idea is to considerate smallest counterexample and prove thet either it can't exist or that there is an even smaller counterexample .

THEOREM Every integer greater than 1 can be written as a product of primes.

Proof:
Suppose the claim is false.
Let nee the smallest integer in M that that cannot be written as a product of primes.
If n is prime, then n is a product of 1 prime.
So n is composite and ∃ k in M. ∃ m in N. (n = mk).
But k and m are seals than n, so they can be written as the product of primes.
Hence n is a product of primes.
This contradicts the definition of n.
Thus by contradiction, the claim is true.

Fun Tutorial 4

Valeria

CSC 240

February 9th 2020

Contents

- Some examples of functions that are not well defined
- Quiz Solution
- problem session on structural induction

Examples of functions that are not well-defined

Explain why each of these proposed recursive definitions of a function from Z^+ to N is not well-defined.

1. $f(n) = \begin{cases} 1 & \text{if } n = 1; \\ 1 + \text{f}(\lfloor (n+1)/2 \rfloor) & \text{if } n \geq 1 \end{cases}$

2. $f(n) = \begin{cases} 0 & \text{if } n = 1; \\ 1 + \text{f(n - 2)} & \text{if } n \geq 2 \end{cases}$

3. $f(n) = \begin{cases} n & \text{if } n \leq 3; \\ 1 + \text{f(n/3)} & \text{if } n \geq 3 \end{cases}$

4. $f(n) = \begin{cases} 1 & \text{if } n = 1; \\ 1 + \text{f(n - 2)} & \text{if } n \text{ is odd}; \\ 1 + \text{f(n/2)} & \text{if } n \text{ is even} \end{cases}$

5. $f(n) = \begin{cases} 2 & \text{if } n = 1; \\ 1 + \text{f(f(n - 1))} & \text{if } n \geq 2 \end{cases}$

Quiz

Recall that R^n is the set of all sequences of n real numbers and R^+ is the set of all finite, nonempty sequences of real numbers.

For any sequence $S \in R^n$, let $-S$ denote the sequence obtained from S by negating each of its elements.

1. Give a recursive definition of the function $max : R^n \to R$ so that for every sequence $S \in R^n$, $max(S)$ is the largest real number that occurs in S.
2. Give a recursive definition of the function $min : R^n \to R$ so that for every sequence $S \in R^n$, $min(S)$ is the smallest real number that occurs in S.
3. Prove by structural induction that $max(-S) = -min(S)$ for every sequence $S \in R^n$.

Quiz Solution

Give a recursive definition of the function $max : R^n \to R$ so that for every sequence $S \in R^n$, $max(S)$ is the largest real number that occurs in S.

Solution:

Base case:

if $x \in R$, then $max(x) = x$

Constructor case:

if $(S, x) \in R^n \times R$, where $n \in Z^+$, then:

$$max(S, x) = \begin{cases} x & \text{if } x > max(S); \\ \text{max(S)} & \text{if } x \leq max(S) \end{cases}$$

Quiz Solution

Give a recursive definition of the function $min : R^{+} \to R$ so that for every sequence $S \in R^{n}$, $min(S)$ is the smallest real number that occurs in S.

Solution:

Base case:

if $x \in R$, then $min(x) = x$

Constructor case:

if $(S, x) \in R^{n} \times R$, where $n \in Z^{+}$, then:

$$min(S, x) = \begin{cases} x & \text{if } x < min(S); \\ \text{min(S)} & \text{if } x \geq min(S) \end{cases}$$

Quiz Solution

Prove by structural induction that $max(-S) = -min(S)$ for every sequence $S \in R^n$.

Solution:
For all $S \in R^n$, let $P(S) =$ "$max(-S) = -min(S)$".

Base Case: If $S \in R$, then, by definition of max, $max(-S) = -S$ and, by the definition of min, $min(S) = S$, so $max(-S) = -S = -min(S)$. Thus $P(S)$ is true.

Constructor Case: Consider $S = (S', x)$, where $S' \in R^n$ and $x \in R$. Note, that $-S = (-S', -x)$. By the induction hypothesis, $max(-S') = -min(S')$.

Quiz Solution

Solution:

For all $S \in R^n$, let $P(S)$ = "$max(-S) = -min(S)$".

Constructor Case: Consider $S = (S', x)$, where $S' \in R^n$ and $x \in R$. Note, that $-S = (-S', x)$. By the induction hypothesis, $max(-S') = -min(S')$.

Case 1: $-x > max(-S')$.

By definition of max, $max(-S) = -x$. In this case $-x > -min(S')$, so $x < min(S')$. By definition of min $min(S) = x$. Thus $max(-S) = -x = -min(S)$.

Case 2: $-x \leq max(-S')$.

Quiz Solution

Solution:
For all $S \in R^n$, let $P(S) =$ "$max(-S) = -min(S)$".

Constructor Case: Consider $S = (S', x)$, where $S' \in R^n$ and $x \in R$. Note, that $-S = (-S', x)$. By the induction hypothesis, $max(-S') = min(S')$.

Case 1: $-x > max(-S')$.
Case 2: $-x \leq max(-S')$.
By definition of max, $max(-S) = max(-S')$. Since $max(-S') = -min(S')$ it follows that $-x \leq -min(S')$, so $x \geq min(S')$. By definition of min, $min(S) = min(S')$. Thus $max(-S) = -min(S) = -min(S)$.

Since one of these two cases holds, $max(-S) = -min(S)$. Thus $P(S)$ is true.
By structural induction, $\forall S \in R^n.P(S)$.

Problem session

1. Strict binary tree is a binary tree in which every node that is not a leaf has exactly 2 children. Give a recursive definition of the set S of strict binary trees.(Recall that a leaf is a node with no children.)

a. Strict binary tree is a binary tree in which every node that is not a leaf has exactly 2 children. Give a recursive definition of the set S of strict binary trees.(Recall that a leaf is a node with no children.)

Solution:

Base Case:

A node is in S.

Constructor Case:

If t_1 and t_2 are in S and if r is a node, then:

b. Link in the chat, discuss in the breakout rooms

Solution:

Problem session

b. Link in the chat, discuss in the breakout rooms

Solution: counterexample. Consider a tree:

Problem session

b. Link in the chat, discuss in the breakout rooms

The mistake is that we cannot apply the induction hypothesis to t', since it is not a strict binary tree. In particular, the node v' has 1 child.

c. Give a recursive definition of the depth of a strict binary tree.

Problem session

c. Give a recursive definition of the depth of a strict binary tree.

For our strict binary tree t, we have: $left(t)$ and $right(t)$ are children of t. We will define $d(t)$ as the depth of a strict binary tree.

Base case:

if t is a node, then $d(t) = 0$.

Constructor case:

if t is not a node, then $d(t) = 1 + max\{d(left(t)), d(right(t))\}$

d. Prove by structural induction that, for all strict binary trees t, $N(t) \leq 2^{d(t)+1} - 1$

Problem session

d. Prove by structural induction that, for all strict binary trees t, $N(t) \leq 2^{d(t)+1} - 1$

For all strict binary trees t, let $P(t) = "N(t) \leq 2^{d(t)+1} - 1"$.

Base case: if t is a node, then $d(t) = 0$, $N(t) = 1$. Therefore, $2^{d(t)+1} - 1 = 1$ and $N(t) \leq 2^{d(t)+1}$. Thus, $P(t)$ is true.
Constructor case: consider a strict binary tree t s.t node r is its root and t_1 and t_2 are its left and right children respectively.
Without loss of generality, assume that $d(t_1) \geq d(t_2)$. There are two things to notice here:

1. $2^{d(t_1)+2} = 2^{d(t_1)+1} + 2^{d(t_1)+1} \geq 2^{d(t_2)+1} + 2^{d(t_1)+1}$.

Problem session

d. Prove by structural induction that, for all strict binary trees t, $N(t) \leq 2^{d(t)+1} - 1$

Constructor case: consider a strict binary tree t s.t node r is its root and t_1 and t_2 are its left and right children respectively.

Without loss of generality, assume that $d(t_1) \geq d(t_2)$. There are two things to notice here:

1. $2^{d(t_1)+2} = 2^{d(t_1)+1} + 2^{d(t_1)+1} \geq 2^{d(t_2)+1} + 2^{d(t_1)+1}$.
2. By definition of $d(t)$, $d(t) = d(t_1) + 1$. Therefore, $d(t_1) + 2 = d(t) + 1$.

We know that $N(t) = N(t_1) + N(t_2) + 1$. By induction hypothesis, $N(t) \leq 2^{d(t_1)+1} - 1 + 2^{d(t_2)+1} - 1 + 1 \leq 2^{d(t_1)+2} - 1 = 2^{d(t)+1} - 1$.

Therefore, $N(t) \leq 2^{d(t)+1} - 1$, which means that $P(t)$ is true. By structural induction, $\forall$ strict binary trees t. $P(t)$.

Solutions to

CSC240 Winter 2021 Homework Assignment 5

1. Consider $H(2,1)$. Line 1 cannot be applied since $1 \neq 0$. Applying line 2 or line 3 gives $H(2,1) = H(1,2)$. Line 1 cannot be applied to $H(1,2)$, since $2 \neq 0$. Line 3 cannot be applied to $H(1,2)$, since $1 < 2$. Therefore, only line 2 can be applied to $H(1,2)$, resulting in $H(1,2) = H(2,1)$. Thus, there is no way to get a value for $H(2,1)$ from the definition.

2. For all natural numbers a and b, let $P(a,b) =$ "$G(a,b)$ has one and only one value."

   Note that, for $b > a$, neither the first line or the third line can be applied to $G(a,b)$, so the only way to define it is by applying the second line and defining it to be the same as $G(b,a)$. Therefore, it suffices to prove $(b \leq a)$ IMPLIES $P(a,b)$

   We will prove $\forall a \in \mathbb{N}.\forall b \in \mathbb{N}.[(b \leq a) \text{ IMPLIES } P(a,b)]$ by double induction on a and b.

   Let $Q(a)$ denote the predicate $\forall b \in \mathbb{N}.[(b \leq a) \text{ IMPLIES } P(a,b)]$.
   We will prove $\forall a \in \mathbb{N}.Q(a)$ by induction on a.

   Let $a \in \mathbb{N}$ be arbitrary. Suppose that $Q(a')$ is true for all $a' \in \mathbb{N}$ such that $a' < a$.

   Let $b \in \mathbb{N}$ be arbitrary and suppose that $b \leq a$. Furthermore, suppose that $P(a,b')$ is true for all $b' \in \mathbb{N}$ such that $b' < b$.

   Case 1: $a = 0$. Since $b \leq a$, it follows that $b = 0$. By line 1, $G(0,0) = 0$. Line 2 says $G(0,0) = G(0,0)$, which is not useful for defining $G(0,0)$. Line 3 cannot be applied. Therefore $G(0,0)$ has only one value and $P(0,b)$ is true.

   Case 2: $a > 0$ and $b = 0$. Then $G(a,0)$ is defined for all $a \in \mathbb{N}$ by line 1. Line 3 cannot be applied. Applying line 2 to $G(a,0)$ results in $G(0,a)$. Since $a > 0$, only line 2 can be applied to $G(0,a)$, which is not useful, since it says that $G(0,a) = G(a,0)$. Therefore, $G(a,0)$ has only one value and $P(a,0)$ is true.

   Case 3: $a > 0$ and $b > 0$. By line 3, $G(a,b) = G(a-b,b)$. Since $b > 0$, line 1 cannot be applied. Applying line 2 to $G(a,b)$ results in $G(b,a)$. Since $a > 0$, line 1 cannot be applied to $G(b,a)$ and, since $a \geq b$, line 3 cannot be applied to $G(b,a)$. Thus, only line 2 can be applied to $G(b,a)$, which results in $G(a,b)$. Therefore applying line 2 to $G(a,b)$ is not useful. Hence only line 3 can be applied.

   Since $b > 0$ and $a \geq b$, it follows that $a - b \in \mathbb{N}$ and $a - b < a$. By assumption, $Q(a-b)$ is true, i.e., $\forall b \in \mathbb{N}.[(b \leq a-b) \text{ IMPLIES } P(a-b,b)]$

   Case 3.1: $a-b \geq b$. By specialization and modus ponens, $P(a-b,b)$ is true. Hence $G(a-b,b)$ has one and only one value, so the same is true for $G(a,b)$. Hence $P(a,b)$ is true.

   Case 3.2: $a-b < b$. Then neither line 1 nor line 3 can be applied, so only line 2 can be applied, resulting in $G(b,a-b)$.

   If $b < a$, then, by assumption, $Q(b)$ is true i.e., $\forall b' \in \mathbb{N}.[(b' \leq b) \text{ IMPLIES } P(b,b')]$. In this case, by specialization and modus ponens, $P(b,a-b)$ is true. Hence $G(b,a-b)$ has one and only one value, so the same is true for $G(a-b,b)$ and $G(a,b)$

   Otherwise, $b = a$, so $G(b,a-b) = G(a,0)$. It follows from Case 2 that $G(a,0)$ has one and only one value, so the same is true for $G(0,a)$ and $G(a,b)$. In both these sub cases, $P(a,b)$ is true.

Hence $P(a,b)$ is true using proof by cases. By direct proof, $(b \leq a)$ IMPLIES $P(a,b)$. It follows by (strong) induction that, $\forall b \in \mathbb{N}.[(b \leq a)$ IMPLIES $P(a,b)]$, which is $Q(a)$. By (strong) induction $\forall a \in \mathbb{N}.Q(a)$ is true.

3. Let $R(a,b)$ denote the predicate "$\exists x \in \mathbb{N}.\exists y \in \mathbb{N}.G(a,b) = xa + yb$".

   Let $(a,b) \in \mathbb{N} \times \mathbb{N}$ be arbitrary.

   If $G(a,b)$ is defined by an application of line 1, then $b = 0$. Since $G(a,0) = a = 1 \cdot a + 1 \cdot 0$, it follows by construction, with $x = 1 = y$, that $R(a,0)$ is true.

   If $G(a,b)$ is defined by an application of line 2 and $R(b,a)$ is true, then $G(a,b) = G(b,a) = xb + ya$ for some $x, y \in \mathbb{N}$, by instantiation. Then, for $x' = y$ and $y' = x$, $G(a,b) = x'a + y'b$. Hence $R(a,b)$ is true.

   If $G(a,b)$ is defined by an application of line 3 and $R(a-b,b)$ is true, then $G(a,b) = G(a-b,b) = x(a-b) + yb$ for some $x, y \in \mathbb{N}$, by instantiation. Then, for $x' = x$ and $y' = y - x$, $G(a,b) = x'a + y'b$. Hence $R(a,b)$ is true.

   By structural induction, $R(a,b)$ is true for all $(a,b) \in \mathbb{N}$.

CSC 240 Lecture 6

A set S is partially ordered if there exists a binary predicate (or relation)
R: S x S → {T,F} such that, for all x,y,z $\in$ S,
R(x,x) = T (reflexive)
(R(x,y) AND R(y,x)) IMPLIES (x = y) (asymmetric)
(R(x,y) AND R(y,z)) IMPLIES R(x,z) (transitive)
R is called a partial order.

Examples
$\leq$ is a partial order for $\mathbb{Z}$ (and $\mathbb{R}$)
$\subseteq$ is a partial order for $\mathscr{P}$({1,2,3})

Not examples
$<$ (not reflexive) for $\mathbb{Z}$
R(x,y) = ``|x| $\leq$ |y|'' since |-1| $\leq$ |1| and |1| $\leq$ |-1| (not asymmetric) for $\mathbb{Z}$

H = hockey teams in a tournament
R(t,t') = T if and only if team t beat team t' (not transitive or reflexive)

S is totally ordered if there exists a partial order R: S x S → {T,F}
such that, for all x,y in S,
R(x,y) OR R(y,x) (comparability)
R is called a total order

example
$\leq$ is a total order for $\mathbb{Z}$ (or $\mathbb{R}$)

not an example
$\subseteq$ is a NOT a total order for $\mathscr{P}$({1,2,3})
since {1,2} $\not\subseteq$ {2,3} and {2,3} $\not\subseteq$ {1,2}

A total order (or partial order) for a set S is a well ordering for S
if every nonempty subset of S has a smallest element.
S is well ordered if there is a well ordering for S.

example: $\leq$ is well-ordering for $\mathbb{N}$.

Note that an empty subset of $\mathbb{N}$ does not have any elements,
so, in particular, it does not have a smallest element.

$\leq$ is not a well-ordering for $\mathbb{Z}$ or $\mathbb{Q}^+$.

But if you consider the ordering by absolute value, and then by value, $\mathbb{Z}$ is well ordered.
0 ,-1,1,-2,2,...
R(x,y) IFF $[(|x| < |y|)$ OR $((|x| = |y|)$ AND $(x \leq y))$]

For $\mathbb{Q}^+$, consider the ordering based on the max{numerator, denominator} when written in reduced form and then order by value.
1/1, 1/2, 2/1, 1/3, 2/3, 3/2, 3/1,1/4, 3/4, 4/3, 4/1,...

Well Ordering Proof
To prove $\forall e \in S.P(e)$, where $\preccurlyeq$ is a well ordering of the set S:

L1. To obtain a contradiction, suppose that $[\forall e \in S.P(e)]$ is false.
L2. Let $C = \{e \in S \mid P(e)$ is false$\}$ be the set of counterexamples to P.

L3. $C \neq \phi$; definition: L1, L2

L4. Let e be the smallest element of C; $\preccurlyeq$ is a well ordering of S, L2, L3

Let $e' = \cdots$

$\vdots$

L5. $e' \in C$.

$\vdots$

L6. $e' \prec e$.

L7. This is a contradiction: L4, L5, L6

$\forall e \in S.P(e)$; proof by contradiction: L1, L7

If $\preccurlyeq$ is a well ordering, we will use $x \prec y$ to mean that $x \preccurlyeq y$ and $x \neq y$.

A rational number $m/n \in \mathbb{Q}^+$ is expressed in reduced form if
$m \in \mathbb{Z}^+$ and $n \in \mathbb{Z}^+$ have no common (prime) factors.

THEOREM Every positive rational number m/n can be expressed in reduced form.

Proof: Suppose there exist $m,n \in \mathbb{Z}^+$ such that m/n cannot be expressed in reduced form.
Let $C = \{ m \in \mathbb{Z}^+ \mid \exists\, n \in \mathbb{Z}^+. (m/n \text{ cannot be expressed in reduced form}) \}$.
Then $C \neq \phi$.

Since $\mathbb{Z}^+$ is well ordered, C has a smallest element, m_0.
By definition of C, there is $n_0 \in \mathbb{Z}^+$ such that m_0 / n_0 cannot be expressed in reduced form.
This means that m_0 and n_0 have a common prime factor $p > 1$.

Let $m'_0 = m_0 / p \in \mathbb{Z}^+$ and $n'_0 = n_0 / p \in \mathbb{Z}^+$.
Since $m'_0 / n'_0 = m_0 / n_0$, it cannot be expressed in reduced form.
Thus $m'_0 \in C$.
But $m'_0 < m_0$.
This contradicts the fact that m_0 is the smallest element of C.

Hence $C = \phi$ and every rational number can be expressed in reduced form.

For every positive integer i
let E(i) denote the subsets of {1,...,i } which contain an even number of elements and
let U(i) denote the subsets of {1,...,i } which contain an odd number of elements.

Theorem For all positive integers i, $|E(i)| = |U(i)| = 2^{i-1}$.

Proof (using the well ordering principle):

For every positive integer i, let P(i) = "$|E(i)| = |U(i)| = 2^{i-1}$ ".

Let $C = \{i \in \mathbb{Z}^+ \mid \text{NOT}(P(i)) \}$.
To obtain a contradiction, suppose that $C \neq \phi$.
Then, by the well ordering principle, C has a smallest element, x.

Note that $x \neq 1$, since {1} has $1 = 2^{x-1}$ subset which contains an even number of elements, namely ϕ, and
1 subset which contains an odd number of elements, namely {1}.

Thus $x > 1$ and P(x-1) is true, so $|E(x-1)| = |U(x-1)| = 2^{x-2}$.

Let $E'(x) = \{ S \in E(x) \mid x \in S \}$.
Since the sets in E(x) that don't contain x are the sets in E(x-1),
it follows that E(x) = E'(x) U E(x-1) and |E(x)| = |E'(x)| + |E(x-1)|.
There is a one-to-one correspondence between E'(x) and U(x-1):
adding x to a set in U(x-1) gives a set in E'(x)
and deleting x from a set in E'(x) gives a set in U(x-1).
Thus |E'(x)| = |U(x-1)|.
Hence $|E(x)| = |U(x-1)| + |E(x-1)| = 2^{x-2} + 2^{x-2} = 2^{x-1}$.

Thus P(x) so $x \notin C$. This is a contradiction.
Hence $C = \phi$ and $\forall i \in \mathbb{Z}^+.P(i)$ is true.

COUNTABLE AND UNCOUNTABLE SETS

A function f:A→B is surjective or onto if and only if $\forall y \in B.\ \exists x \in A.\ f(x) = y$.
Each element of B is mapped to by some element of A.

From the existence of a surjective function f:A→B,
where A and B are finite sets, we can conclude that $|A| \geq |B|$

A nonempty set C is called countable if
there is a surjective function from $\mathbb{N}$ to C.

Every nonempty finite set is countable.

Suppose C is a finite set.
Arrange the elements in some order: $c_0\ c_1, \ldots, c_{x-1}$
Define $f:\mathbb{N} \rightarrow C$ as follows:

$$f(i) = \begin{cases} c_i & \text{for } i = 0,\ldots,x-1 \\ c_{x-1} & \text{for } i \geq x \end{cases}$$

Problem: what if C is empty? There is no such function f,
since there is no place to map to.
We also define the empty set to be countable.

$\mathbb{Z}$ is countable
0,-1,1,-2,2

$$f(n) = \begin{cases} n/2 & \text{if } n \text{ is even} \\ -(n+1)/2 & \text{if } n \text{ is odd} \end{cases}$$

More generally,
if A and B are countable, then so is A U B.

If A is countable and B $\subseteq$ A, then B is countable.

Example: the set of odd integers is countable.

If A and B are countable, then A x B = $\{(a,b) \mid (a \in A) \text{ AND } (b \in B)\}$
is countable.

For example, $\mathbb{N} \times \mathbb{N}$ is countable.

f	0	1	2	3 ...
0	0	1	3	6
1	2	4	7	
2	5	8		
3	9			
⋮				

f(k) = (x,y) if location (x,y) of this infinite array contains the number k.

Lemma If A is nonempty and countable and there is a surjective function $f:A \to B$, then B is countable.

Proof:
Since A is countable, there is a surjective function $g:\mathbb{N} \to A$.
Then there is a surjective function $h:\mathbb{N} \to B$
defined by $h(i) = f(g(i))$ for all $i \in \mathbb{N}$.

$\mathbb{Q}^+$ is countable
$f: \mathbb{Z}^+ \times \mathbb{Z}^+ \to \mathbb{Q}^+$
$f(a,b) = a/b$
Note that f is not 1 to 1: every element of $\mathbb{Q}^+$ has an infinite number of pairs that are mapped to it by f.
For example, (1,2), (2,4), (3,6) all map to 1/2

$\{0,1\}^*$ = the set of all finite binary sequences is countable:
λ, 0, 1, 00, 01, 10, 11, 000, ...

Consider the surjective function $g: \mathbb{N} \times \mathbb{N} \to \{0,1\}^*$ where

$$g(i,j) = \begin{cases} j\text{'th lexicographically smallest string of length } i \text{ if } 1 \le j \le 2^i \\ \lambda, \text{ otherwise.} \end{cases}$$

There are 2^i binary strings of length i.

The set of all finite strings of ASCII characters is countable.

The set of all syntactically correct Python programs is countable.

For any set A, the power set of A is the set of all subsets of A
i.e. $\mathscr{P}(A) = \{ S \mid S \subseteq A \}$.

If A has size n, what is the size of $\mathscr{P}(A)$? 2^n

Theorem $\mathscr{P}(\mathbb{N})$ is uncountable.
(MIT text Chapter 8, Corollary 8.1.13, section 8.1.4)

Proof: Suppose $\mathscr{P}(\mathbb{N})$ is countable.

Then there is a surjective function $f: \mathbb{N} \to \mathscr{P}(\mathbb{N})$.
Let $D = \{ i \in \mathbb{N} \mid i \notin f(i) \} \in \mathscr{P}(\mathbb{N})$.

Since f is surjective, there exists $j \in \mathbb{N}$ such that $f(j) = D$.
Then, for all $i \in \mathbb{N}$,

$i \in f(j)$ IFF $i \in D$, since $f(j) = D$
IFF $i \notin f(i)$, by definition of D.

Since $j \in \mathbb{N}$, by specialization,
$j \in f(j)$ IFF $j \notin f(j)$.
This is a contradiction.
Therefore $\mathscr{P}(\mathbb{N})$ is uncountable.

This is an example of a proof by diagonalization.

Given any $S \subseteq \{1,2,3,4\}$
we can represent it by the binary sequence S_1, S_2, S_3, S_4 of length 4,
where

$$S_i = \begin{cases} 1 \text{ if } i \in S \\ 0 \text{ if } i \notin S. \end{cases}$$

for example
0,1,1,0 denotes the set {2,3}
0,0,0,0 denotes the empty set.
This sequence is called the characteristic vector of the set.

Likewise, given any $S \subseteq \mathbb{N}$,
we can represent it by the infinite binary sequence
$S_0, S_1, S_2, \ldots$

where

$$S_i = \begin{cases} 1 \text{ if } i \in S \\ 0 \text{ if } i \notin S. \end{cases}$$

If f is surjective, then
f(0),
f(1),
⋮
is a list of all subsets of $\mathbb{N}$, possibly with duplications.

Consider the characteristic vectors of all of these sets.
$f(0)_0, f(0)_1, \ldots$
$f(1)_0, f(1)_1, \ldots$
⋮

This is an infinite 2 dimensional Boolean array M
where $M[i,j] = f(i)_j$
In other words,

$$M[i,j] = \begin{matrix} 1 \text{ if } j \in f(i) \\ 0 \text{ if } j \notin f(i) \end{matrix}$$

M	0	1	2	3...
0	1	0	0	0
1	0	0	0	0
2	1	0	1	0
3	0	0	0	0
⋮				

The rows and columns are indexed by the natural numbers.
Row i represents the i'th set in our enumeration.
Note that every set is guaranteed to occur in the list,
but the same set can occur in the list f(0), f(1), f(2),... many times,
since f is surjective (onto), but not necessarily injective (1-to-1).

For example,
f(0) could be {0},
f(1) could be the empty set
f(2) could be the set of all even natural numbers
f(3) could be the empty set

Consider the set $D = \{ i \in \mathbb{N} \mid i \notin f(i) \} \in \mathscr{P}(\mathbb{N})$

Note that the characteristic vector of D is
the complement of the diagonal of M.

The set D can be constructed by going along the diagonal:
put i in D if and only if M[i,i] = 0.

By doing so, we ensure that $D \neq f(i)$ for all $i \in \mathbb{N}$.

More generally,

THEOREM 8.1.12 Cantor's theorem

For any set A, there is no surjective function from A to $\mathscr{P}(A)$.

Another example of a diagonalization proof
Halting problem (MIT book Section 8.2)

There is a compiler (which is a program) G that determines whether a given ASCII string P is a syntactically correct Python function that takes a string as input

i.e. it that takes a single ASCII string P as input,
outputs True if P is a syntactically correct Python function that takes a string as input
and, outputs False if P is not a syntactically correct Python function.
It is easy to modify a Python compiler to do this.

It would be nice to have a Python program
H that takes as input two strings, P and x, and decides whether
P is a syntactically correct Python function that takes one string as input
 and eventually returns (i.e.halts) on input x.
i.e. H(P,x) returns True if
P is a syntactically correct Python function that takes one string as input
and returns on input x
and H(P,x) returns False otherwise, i.e. if P is not syntactically correct,
or P doesn't take one string as its input, or if P runs forever (gets into an infinite loop) on input x.

```
def H(P:str, x:str):
⋮
```

```
>>> justreturn = 'def j(s: str): return s'
>>> H(justreturn, 'hello')
True
```

```
>>> goforever = 'def g(s: str): while True: pass'
>>> H(goforever, 'hello')
False
```

The halting problem is to construct such a program H.
Unfortunately, the halting problem is unsolvable.

To obtain a contradiction, suppose that such a program H exists.
Consider the syntactically correct Python function D that takes one string as input:

```
def D(x:str):
  if H(x,x):
     while True: pass
  else: return True
```

```
>>> d = 'def D(x):\n if H(x,x): \n\t while True: pass\n else: return True'
```

What happens when D runs on input d?

```
>>> D(d)
```


from the code of D:
if H(d,d) = False then D(d) returns
if H(d,d) = True then D(d) goes into an infinite loop

from the definition of H:
if D(d) returns, then H(d,d) = True
if D(d) goes into an infinite loop, then H(d,d) = False

There is a contradiction, no matter which case occurs:
if H(d,d) = False then D(d) returns, so H(d,d) = True
if H(d,d) = True then D(d) goes into an infinite loop, so H(d,d) = False.
Thus the halting problem is unsolvable.

Pictorially, we can view the function H(P,x) using an infinite matrix whose rows and columns are indexed by the set of all ASCII strings, which is countable.

		x			
	H	λ	a	b	c...
	λ	T	F	F	F
P	a	F	F	F	F
	b	T	F	T	F
	c	F	F	F	F
	⋮				

If H(x,x) = False then D(x) halts, so H(d,x) = True
If H(x,x) = True then D(x) runs forever, so H(d,x) = False

Thus row d is the complement of the diagonal of the matrix H.

In particular, element d of row d is the complement of element d of the diagonal.

This is a contradiction, since this is the same element.

Reductions

THEOREM There is no Python program A that takes as input a string P and decides whether P is a syntactically correct Python function that takes one string as input and eventually returns (i.e.halts) on every input.
i.e. A(P) returns True if P is syntactically correct, takes 1 string as input, and returns on all inputs and
A(P) returns False otherwise.

Proof:
To obtain a contradiction, suppose there is such a program A.

Let $\mathscr{C}$ be the set of all syntactically syntactically correct Python functions that take one string as input.

Let H' be the Python program that takes two strings P and x as input and behaves as follows:
if P $\notin \mathscr{C}$, then return False.
Otherwise, let Q be constructed from P by adding a new first line
s = x
to its program, where s is the input variable for program P and x is the other input to H'.
output (A(Q))

For example, recall the string
justreturn = 'def j(s: str): return s'

Then H'(justreturn,'hi')
determines that justreturn $\in \mathscr{C}$,
it constructs Q = 'def j(s:str):\n s = 'hi' \n return s',
it runs A on input Q, which returns True,
and returns True.

Similarly, if
goforever = 'def g(s: str): while True: pass'
Then H'(goforever,'hi')
determines that goforever $\in \mathscr{C}$,
it constructs Q = 'def g(s:str):\n s = 'hi' \n while True: pass',
it runs A on input Q, which returns False,
and returns False.

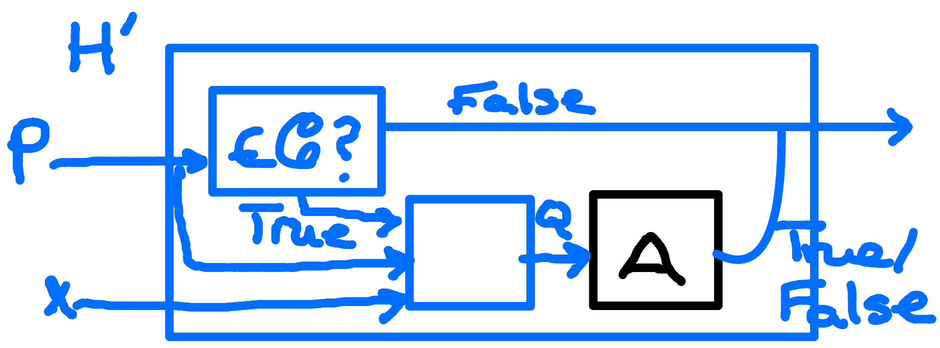
Claim: H' solves the halting problem:

Proof:
If P $\notin \mathscr{C}$, then H'(P,x) outputs False = H(P,x).

So suppose P $\in \mathscr{C}$.
If P halts on input x,
then Q halts on all inputs, so A(Q) returns True
and hence H'(P,x) returns True = H(P,x).

If P doesn't halt on input x,
then, on all inputs, Q doesn't halt, so A(Q) returns False
and hence H'(P,x) returns False = H(P,x).

Hence H' solves the halting problem.
But the halting problem is unsolvable.
Hence there is no such program A.



Asymptotic Notation

Let $\mathscr{F}$ denote the set of all functions from $\mathbb{N}$ to $\mathbb{R}^+ \cup \{0\}$,
from the natural numbers to the nonnegative real numbers.
For any function $f \in \mathscr{F}$, let

$O(f) = \{g \in \mathscr{F} \mid \exists c \in \mathbb{R}^+.\ \exists b \in \mathbb{N}.\ \forall n \in \mathbb{N}.\ [(n \geq b) \text{ IMPLIES } (g(n) \leq c\, f(n))]\}$.

$g \in O(f)$ means that,
for all n sufficiently large (i.e. for all but finitely many $n \in \mathbb{N}$),
$g(n)$ is at most a constant factor times $f(n)$.

Example:

$6n+4$ is in $O(3n)$ since $6n + 4 \leq 3 \times 3n$ for all $n \geq 2$.
Here $c=3$ and $b=2$.
We could also have chosen $c=4$ and $b=1$.
If there is one choice for c and b that work,
then there are an infinite number of choices that work.

It is customary in computer science to write
$O(n^2)$ instead of
$O(f)$, where $f(n) = n^2$ for all $n \in \mathbb{N}$.
In other words, we consider n^2 to be a function from $\mathbb{N}$ to $\mathbb{N}$.

$\Omega(f) = \{g \in \mathscr{F} \mid \exists c \in \mathbb{R}^+.\ \exists b \in \mathbb{N}.\ \forall n \in \mathbb{N}.\ [(n \geq b) \text{ IMPLIES } (g(n) \geq c\, f(n))]\}$

$g \in \Omega(f)$ means that,
for all n sufficiently large (i.e. for all but finitely many $n \in \mathbb{N}$),
$g(n)$ is at least a constant factor times $f(n)$.

$(g \in \Omega(f))$ IFF $(f \in O(g))$

$\theta(f) = \{g \in \mathscr{F} \mid \exists c \in \mathbb{R}^+.\ \exists c' \in \mathbb{R}^+. \exists b \in \mathbb{N}.\ \forall n \in \mathbb{N}.$
$\quad [(n \geq b) \text{ IMPLIES } (cf(n) \leq g(n) \leq c'f(n))]\}$

$= O(f) \cap \Omega(f)$

PROPERTIES OF O NOTATION

(see summary in Course Materials.)

THEOREM The set of all functions from $\mathbb{N}$ to $\mathbb{N}$ is uncountable.

Let $\mathcal{F}$ be the set of all functions from $\mathbb{N}$ to $\mathbb{N}$.
To obtain a contradiction, assume $\mathcal{F}$ is countable.
Then there is a surjective function f from $\mathbb{N}$ to $\mathcal{F}$.

THEOREM The set of all functions from $\mathbb{N}$ to $\mathbb{N}$ is uncountable.

Let $\mathcal{F}$ be the set of all functions from $\mathbb{N}$ to $\mathbb{N}$.
To obtain a contradiction, assume $\mathcal{F}$ is countable.
Then there is a surjective function f from $\mathbb{N}$ to $\mathcal{F}$.

Let g be defined so that, for all $n \in \mathbb{N}$, $g(n) = f_n(n) + 1$.
Since $f_n : \mathbb{N} \to \mathbb{N}$, it follows that $f_n(n) \in \mathbb{N}$.
Since $\mathbb{N}$ is closed under addition, $g(n) = f_n(n) + 1 \in \mathbb{N}$.
Thus $g \in \mathcal{F}$.

THEOREM The set of all functions from $\mathbb{N}$ to $\mathbb{N}$ is uncountable.

Let $\mathcal{F}$ be the set of all functions from $\mathbb{N}$ to $\mathbb{N}$.
To obtain a contradiction, assume $\mathcal{F}$ is countable.
Then there is a surjective function f from $\mathbb{N}$ to $\mathcal{F}$.
Let g be defined so that, for all $n \in \mathbb{N}$, $g(n) = f_n(n) + 1$.
Since $f_n : \mathbb{N} \to \mathbb{N}$, it follows that $f_n(n) \in \mathbb{N}$.
Since $\mathbb{N}$ is closed under addition, $g(n) = f_n(n) + 1 \in \mathbb{N}$.
Thus $g \in \mathcal{F}$.
Since f is surjective, there exists $n \in \mathbb{N}$ such that $g = f_n$.
By definition, $g(n) = f_n(n) + 1 \neq f_n(n) = g(n)$.
This is a contradiction. Thus $\mathcal{F}$ is uncountable.

THEOREM The set of all functions from $\mathbb{N}$ to $\mathbb{N}$ is uncountable.

Let $\mathcal{F}$ be the set of all functions from $\mathbb{N}$ to $\mathbb{N}$.
To obtain a contradiction, assume $\mathcal{F}$ is countable.
Then there is a surjective function f from $\mathbb{N}$ to $\mathcal{F}$.

Let g be defined so that, for all $n \in \mathbb{N}$, $g(n) = f_n(n) + 1$.
Since $f_n : \mathbb{N} \to \mathbb{N}$, it follows that $f_n(n) \in \mathbb{N}$.
Since $\mathbb{N}$ is closed under addition, $g(n) = f_n(n) + 1 \in \mathbb{N}$.
Thus $g \in \mathcal{F}$.

Since f is surjective, there exists $n \in \mathbb{N}$ such that $g = f_n$.
By definition, $g(n) = f_n(n) + 1 \neq f_n(n) = g(n)$.
This is a contradiction. Thus $\mathcal{F}$ is uncountable.

	0	1	2	$\cdots$
f_0	$f_0(0)$	$f_0(1)$	$f_0(2)$	
f_1	$f_1(0)$	$f_1(1)$	$f_1(2)$	
f_2	$f_2(0)$	$f_2(1)$	$f_2(2)$	
$\vdots$				

Properties of Big-O Notation

① Constant factors don't matter

If $d > 0$ is a constant

$d \cdot f(n) \in O(f(n))$

$f(n) \in O(d(f(n)))$

$f(n) = \frac{n^2}{2}$ $\qquad g(n) = 2n^2$ $\qquad f(n) = O(g(n))$

$g(n) = O(f(n))$

$f(x) \in O(g(x))$ if

$f(x) \leq c \cdot g(x)$ for all $x > N$

$\leq c \cdot d \cdot f(x)$

$1/d$

② Low-order terms don't matter

If $\lim_{n \to \infty} \frac{h(n)}{g(n)} = 0$

then $g(n) + h(n) \in O(g(n))$

Ex $n^2 + n \in O(n^2)$ $\qquad$ Eventually $h(n) \leq g(n)$

$g(n) + h(n) \leq 2 \cdot g(n)$ for n large enough

③ Transitivity

If $f(n) \in O(g(n))$ and $g(n) \in O(h(n))$

then $f(n) \in O(h(n))$

$f(n) \leq C_1 \cdot g(n)$ if $n > N_1$ $\qquad g(n) \leq C_2 \cdot h(n)$ if $n > N_2$

$\Rightarrow f(n) \leq C_1 \cdot C_2 \cdot h(n)$ if $n > \max\{N_1, N_2\}$

④ Summation Rules

If $f_1(n) \in O(g_1(n))$ and $f_2(n) \in O(g_2(n))$
then $f_1(n) + f_2(n) \in O(g_1(n) + g_2(n))$

$f_1(n) + f_2(n) \in O(\max\{f_1, f_2\})$
same goes for g_1, g_2

⑤ Product Rules.

If $f_1(n) \in O(g_1(n))$ and $f_2(n) \in O(g_2(n))$
then $f_1(n) \times f_2(n) \in O(g_1(n) \times g_2(n))$

⑥ Exponents do matter
- if $a \leq b$ then $n^a \in O(n^b)$

(Let $b = a + \varepsilon$, then $n^b = n^a n^\varepsilon$)
- if $a > b$ then $n^a \notin O(n^b)$

⑦ Base of exponents matter
- if $1 < a \leq b$ then $a^n \in O(b^n)$

<u>Ex</u> $\sqrt{2}^n$ vs 2^n
$\sqrt{2}^n = 2^{n/2}$ $\quad (\sqrt{2} = 2^{1/2})$

Follows from same logic as point ⑥
- if $1 < b < a$ then $a^n \notin O(b^n)$

⑧ Base of logarithms don't matter
for all $a, b > 1$
$\log_a(n) \in O(\log_b(n))$
$\log_a(b) = \dfrac{\log_c(b)}{\log_c(a)}$ for any c

$$\log_a(n) = \frac{\log_b(n)}{\log_b(a)} \leftarrow \text{constant}$$

(9) Exponential functions grow faster than polynomial functions for all $b>1$ and all a

- $n^a \in O(b^n)$
- $b^n \notin O(n^a)$

Ex $n^{1,000,000} \in O(1.001^n)$

(10) Polynomial functions grow faster than polylog functions for all $a, b > 0$

$(\log n)^a \in O(n^b)$

$n^b \notin O((\log n)^a)$

Ex $n^{0.000001} \notin O((\log n)^{1,000,000})$

$\mathbb{R}$ is not Countable

To obtain a contradiction assume $\mathbb{R}$ is countable

Assume $[0,1] \cap \mathbb{R}$ is countable (follows from above)

(real numbers between 0 and 1)

There is a surjective function $D: \mathbb{N} \to \mathbb{R}$

$D(j) = 0.\_\ \_\ \_\ \_\ \_\ \_\ \_\ \_\ \_\ \_ \ldots$ with $d_{j,1}$, $d_{j,2}$, $d_{j,i}$

$D(j) = 0$	write as	$0.000000\ldots$
$D(j) = 1/3$		$0.33333\ldots$
$D(j) = 1$		$0.99999\ldots$
$D(j) = 1/4$		$0.25000\ldots$

$c = 0.c_1c_2\ldots$ where for all i $c_i = \begin{cases} d_{i,i}+2 & \text{if } d_{i,i} < 5 \\ d_{i,i}-2 & \text{if } d_{i,i} \geq 5 \end{cases}$

Ex $D(2) = 1/2 = 0.5000000$

$c_2 = 2\ (0+2)$ $\quad c \neq D(j)$ for all $j \in \mathbb{N}$.

Assume $c = D(j)$ for some j.

But by def of c, $c_j \neq D(j)_j$: a contradiction!

Question: Why can't this proof be used to prove that $\mathbb{Q}$ is uncountable?

Answer: c may not be in $\mathbb{Q}$.

In fact, this gives us a proof of the existence of irrational numbers.

- $\mathbb{Q}$ is countable

- Construct digit representation as above
- Construct c as above

- Then $c \notin \mathbb{Q}$, but c is a real number!
- Therefore c is irrational

Solutions to

CSC240 Winter 2021 Homework Assignment 6

1. This is known as Kraft's Inequality.

   Let $P : \mathcal{B} \to \{\mathrm{T}, \mathrm{F}\}$ be the predicate such that, for every binary tree $T \in \mathcal{B}$, $P(T) =$ " $\sum_{x \in leaves(T)} 2^{-depth(T,x)} \leq 1$".

   To obtain a contradiction, assume that $\forall T \in \mathcal{B}.P(T)$ is false.
   Let $\mathcal{C} = \{T \in \mathcal{B} \mid \text{ NOT } (P(T))\}$ be the set of counterexamples to P.
   By assumption, $\mathcal{C} \neq \phi$. Since $\preceq$ is a well ordering of $\mathcal{B}$, the well ordering principle implies that $\mathcal{C}$ has a smallest element, T.

   If T is the empty binary tree, then it has no leaves, so

   $$\sum_{x \in leaves(T)} 2^{-depth(T,x)} = 0 \leq 1.$$

   Thus T is not the empty binary tree.

   If T consists of a single node, then it has one leaf of depth 0, so

   $$\sum_{x \in leaves(T)} 2^{-depth(T,x)} = 2^{-0} = 1.$$

   Therefore T contains at least two nodes.

   Let $L = T$.left and $R = T$.right be the left and right subtrees of T. If $P(L)$ and $P(R)$ are both true, then

   $$\sum_{x \in leaves(L)} 2^{-depth(L,x)} \leq 1 \text{ and } \sum_{x \in leaves(R)} 2^{-depth(R,x)} \leq 1.$$

   Note that, for every node x in L, $depth(T,x) = depth(L,x) + 1$, since the unique path from x to the root of T consists of the unique path from x to the root of L followed by the edge from the root of L to the root of T. Similarly, for every node x in R, $depth(T,x) = depth(R,x) + 1$.

   Since the root of T is not a leaf, $leaves(T)$ is the disjoint union of $leaves(L)$ and $leaves(R)$. It follows that

   $$\begin{aligned}
   \sum_{x \in leaves(T)} 2^{-depth(T,x)} &= \sum_{x \in leaves(L)} 2^{-depth(T,x)} + \sum_{x \in leaves(R)} 2^{-depth(T,x)} \\
   &= \sum_{x \in leaves(L)} 2^{-depth(L,x)-1} + \sum_{x \in leaves(R)} 2^{-depth(R,x)-1} \\
   &= \frac{1}{2} \sum_{x \in leaves(L)} 2^{-depth(L,x)} + \frac{1}{2} \sum_{x \in leaves(R)} 2^{-depth(R,x)} \\
   &\leq \frac{1}{2} \cdot 1 + \frac{1}{2} \cdot 1 \\
   &= 1.
   \end{aligned}$$

Thus $P(T)$ is true. This is a contradiction. Hence, at least one of $P(L)$ and $P(R)$ is false and at least one of L and R is in $\mathcal{C}$.

But $L \preceq T$ and $R \preceq T$, contradicting the definition of T. Thus $\mathcal{C} = \phi$ and $\forall T \in \mathcal{B}.P(T)$ is true.

2. To obtain a contradiction, suppose there is such a program E.

   Let $\mathcal{C}$ be the set of all syntactically correct Python functions that takes one string as input.

   Let H' be the Python program that takes two strings P and x as input and behaves as follows: if $P \in \mathcal{C}$, then return False. Otherwise, let P' be constructed from P by adding a new first line $s = x$ to its program, where s is the input variable for function P and x is the other input to H', and changing all the return statements of P to return True. Let $Q =$ 'def j(s: str): return True'. Finally, H' runs $E(P', Q)$ and outputs the result.

   Note that Q halts and returns True on every input y.

   If $P \notin \mathcal{C}$, then $H'(P, x)$ outputs False.

   If P halts on input x, then P' halts and returns True on every input y. Thus $E(P', Q)$ returns True and $H'(P, x)$ returns True.

   If P doesn't halt on input x, then P' doesn't halt on any input y. Thus $E(P', Q)$ returns False and $H'(P, x)$ returns False.

   Hence $H'(P, x)$ solves the halting problem. This is impossible. Therefore, there is no such program E.

Proof Outlines

LINE NUMBERS: Only lines that are referred to have labels (for example, L1) in this document. For a formal proof, all lines are numbered. Line numbers appear at the beginning of a line. You can indent line numbers together with the lines they are numbering or all line numbers can be unindented, provided you are consistent.

INDENTATION: Indent when you make an assumption or define a variable. Unindent when this assumption or variable is no longer being used.

1. **Implication**: Direct proof of A IMPLIES B.

    L1. Assume A.

    ⋮

    L2. B

   A IMPLIES B; direct proof: L1, L2

2. **Implication**: Indirect proof of A IMPLIES B.

    L1. Assume $NOT(B)$.

    ⋮

    L2. $NOT(A)$

   A IMPLIES B; indirect proof: L1, L2

3. **Equivalence**: Proof of A IFF B.

    L1. Assume A.

    ⋮

    L2. B

   L3. A IMPLIES B; direct proof: L1, L2

    L4. Assume B.

    ⋮

    L5. A

   L6. B IMPLIES A; direct proof: L4, L5

   A IFF B; equivalence: L3, L6

4. **Proof by contradiction** of A.

    L1. To obtain a contradiction, assume $NOT(A)$.

    ⋮

        L2. B

    ⋮

        L3. $NOT(B)$

    L4. This is a contradiction: L2, L3

   Therefore A; proof by contradiction: L1, L4

5. **Modus Ponens**.

   $\vdots$
   L1. A
   $\vdots$
   L2. A IMPLIES B
   B; modus ponens: L1, L2

6. **Conjunction:** Proof of A AND B:

   $\vdots$
   L1. A
   $\vdots$
   L2. B
   A AND B; proof of conjunction; L1, 2

7. **Use of Conjunction**:

   $\vdots$
   L1. A AND B
   A; use of conjunction: L1
   B; use of conjunction: L1

8. **Implication with Conjunction**: Proof of (A_1 AND A_2) IMPLIES B.

   L1. Assume A_1 AND A_2.
   A_1; use of conjunction, L1
   A_2; use of conjunction, L1
   $\vdots$
   L2. B
   (A_1 AND A_2) IMPLIES B; direct proof, L1, L2

9. **Implication with Conjunction**: Proof of A IMPLIES (B_1 AND B_2).

   L1. Assume A.
   $\vdots$
   L2. B_1
   $\vdots$
   L3. B_2
   L4. B_1 AND B_2; proof of conjunction: L2, L3
   A IMPLIES (B_1 AND B_2); direct proof: L1, L4

10. **Disjunction**: Proof of A OR B and B OR A.

   $\vdots$
   L1. A
   A OR B; proof of disjunction: L1
   B OR A; proof of disjunction: L1

11. **Proof by cases**.

L1. C OR $NOT(C)$ tautology
L2. Case 1: Assume C.

$\vdots$

L3. A

L4. C IMPLIES A; direct proof: L2, L3
L5. Case 2: Assume $NOT(C)$.

$\vdots$

L6. A

L7. $NOT(C)$ IMPLIES A; direct proof: L5, L6
A proof by cases: L1, L4, L7

12. **Proof by cases** of A OR B.

L1. C OR $NOT(C)$ tautology
L2. Case 1: Assume C.

$\vdots$

L3. A
L4. A OR B; proof of disjunction, L3

L5. C IMPLIES (A OR B); direct proof, L2, L4
L6. Case 2: Assume $NOT(C)$.

$\vdots$

L7. B
L8. A OR B; proof of disjunction, L7

L9. $NOT(C)$ IMPLIES (A OR B); direct proof: L6, L8
A OR B; proof by cases: L1, L5, L9

13. **Implication with Disjunction**: Proof by cases of (A_1 OR A_2) IMPLIES B.

L1. Case 1: Assume A_1.

$\vdots$

L2. B

L3. A_1 IMPLIES B; direct proof: L1,L2
L4. Case 2: Assume A_2.

$\vdots$

L5. B

L6. A_2 IMPLIES B; direct proof: L4, L5
(A_1 OR A_2) IMPLIES B; proof by cases: L3, L6

14. **Implication with Disjunction**: Proof by cases of A IMPLIES $(B_1$ OR $B_2)$.

   L1. Assume A.
   L2. C OR NOT(C) tautology
   L3. Case 1: Assume C.
   $\vdots$
   L4. B_1
   L5. B_1 OR B_2; disjunction: L4
   L6. C IMPLIES $(B_1$ OR $B_2)$; direct proof: L3, L5
   L7. Case 2: AssumeNOT(C).
   $\vdots$
   L8. B_2
   L9. B_1 OR B_2; disjunction: L8
   L10. NOT(C) IMPLIES $(B_1$ OR $B_2)$; direct proof: L7, L9
   L11. B_1 OR B_2; proof by cases: L2, L6, L10
   A IMPLIES $(B_1$ OR $B_2)$: direct proof. L1, L11

15. **Substitution of a Variable in a Tautology**:
   Suppose P is a propositional variable, Q is a formula, and R' is obtained from R by replacing *every* occurrence of P by (Q).

   L1. R tautology
   R'; substitution of all P by Q: L1

16. **Substitution of a Formula by a Logically Equivalent Formula**:
   Suppose S is a subformula of R and R' is obtained from R by replacing *some* occurrence of S by S'.

   L1. R
   L2. S IFF S'
   L3. R'; substitution of an occurrence of S by S': L1, L2

17. **Specialization**:

   L1. $c \in D$
   L2. $\forall x \in D.P(x)$
   $P(c)$; specialization: L1, L2

18. **Generalization**: Proof of $\forall x \in D.P(x)$.

   L1. Let x be an arbitrary element of D.
   $\vdots$
   L2. $P(x)$
   Since x is an arbitrary element of D,
   $\forall x \in D.P(x)$; generalization: L1, L2

19. **Universal Quantification with Implication**: Proof of $\forall x \in D.(P(x) \text{ IMPLIES } Q(x))$.

- L1. Let x be an arbitrary element of D.
  - L2. Assume $P(x)$
  - $\vdots$
  - L3. $Q(x)$
- L4. $P(x)$ IMPLIES $Q(x)$; direct proof: L2, L3

Since x is an arbitrary element of D,
$\forall x \in D.(P(x) \text{ IMPLIES } Q(x))$; generalization: L1, L4

20. **Implication with Universal Quantification**: Proof of $(\forall x \in D.P(x))$ IMPLIES A.

- L1. Assume $\forall x \in D.P(x)$.
- $\vdots$
- L2. $a \in D$
- $P(a)$; specialization: L1, L2
- $\vdots$
- L3. A

Therefore $(\forall x \in D.P(x))$ IMPLIES A; direct proof: L1, L3

21. **Implication with Universal Quantification**: Proof of A IMPLIES $(\forall x \in D.P(x))$.

- L1. Assume A.
  - L2. Let x be an arbitrary element of D.
  - $\vdots$
  - L3. $P(x)$
- Since x is an arbitrary element of D,
- L4. $\forall x \in D.P(x)$; generalization, L2, L3

A IMPLIES $(\forall x \in D.P(x))$; direct proof: L1, L4

22. **Instantiation**:

L1. $\exists x \in D.P(x)$
- Let $c \in D$ be such that $P(c)$; instantiation: L1
- $\vdots$

23. **Construction**: Proof of $\exists x \in D.P(x)$.

- L1. Let $a = \cdots$
- $\vdots$
- L2. $a \in D$
- $\vdots$
- L3. $P(a)$

$\exists x \in D.P(x)$; construction: L1, L2, L3

24. **Existential Quantification with Implication**: Proof of $\exists x \in D.(P(x) \text{ IMPLIES } Q(x))$.

L1. Let $a = \cdots$
$\vdots$
L2. $a \in D$
L3. Suppose $P(a)$.
$\vdots$
L4. $Q(a)$
L5. $P(a)$ IMPLIES $Q(a)$; direct proof: L3, L4
$\exists x \in D.(P(x) \text{ IMPLIES } Q(x))$; construction: L1, L2, L5

25. **Implication with Existential Quantification**: Proof of $(\exists x \in D.P(x))$ IMPLIES A.

L1. Assume $\exists x \in D.P(x)$.
Let $a \in D$ be such that $P(a)$; instantiation: L1
$\vdots$
L2. A
$(\exists x \in D.P(x))$ IMPLIES A; direct proof: L1, L2

26. **Implication with Existential Quantification**: Proof of A IMPLIES $(\exists x \in D.P(x))$.

L1. Assume A.
L2. Let $a = \cdots$
$\vdots$
L3. $a \in D$
$\vdots$
L4. $P(a)$
L5. $\exists x \in D.P(x)$; construction: L2, L3, L4
A IMPLIES $(\exists x \in D.P(x))$; direct proof: L1, L5

27. **Subset**: Proof of $A \subseteq B$.

L1. Let $x \in A$ be arbitrary.
$\vdots$
L2. $x \in B$
The following line is optional:
L3. $x \in A$ IMPLIES $x \in B$; direct proof: L1, L2
$A \subseteq B$; definition of subset: L3 (or L1, L2, if the optional line is missing)

28. **Weak Induction**: Proof of $\forall n \in N.P(n)$

Base Case:

$\vdots$

L1. $P(0)$

  L2. Let $n \in N$ be arbitrary.

    L3. Assume $P(n)$.

    $\vdots$

    L4. $P(n+1)$

  The following two lines are optional:

  L5. $P(n)$ IMPLIES $(P(n+1)$; direct proof of implication: L3, L4

L6. $\forall n \in N.(P(n)$ IMPLIES $P(n+1))$; generalization L2, L5

$\forall n \in N.P(n)$ induction; L1, L6 (or L1, L2, L3, L4, if the optional lines are missing)

29. **Strong Induction**: Proof of $\forall n \in N.P(n)$

  L1. Let $n \in N$ be arbitrary.

    L2. Assume $\forall j \in N.(j < n$ IMPLIES $P(j))$

    $\vdots$

    L3. $P(n)$

  The following two lines are optional:

  L4. $\forall j \in N.(j < n$ IMPLIES $P(j))$ IMPLIES $P(n)$; direct proof of implication: L2, L3

L5. $\forall n \in N.[\forall j \in N.(j < n$ IMPLIES $P(j))$ IMPLIES $P(n)]$; generalization: L1, L4

$\forall n \in N.P(n)$; strong induction: L5 (or L1, L2, L3, if the optional lines are missing)

30. **Structural Induction**: Proof of $\forall e \in S.P(e)$, where S is a recursively defined set

Base case(s):

  L1. For each base case e in the definition of S

  L2. $P(e)$.

Constructor case(s):

  L3. For each constructor case e of the definition of S,

    L4. assume $P(e')$ for all components e' of e.

    $\vdots$

    L5. $P(e)$

$\forall e \in S.P(e)$; structural induction: L1, L2, L3, L4, L5

31. **Well Ordering Principle**: Proof of $\forall e \in S.P(e)$, where S is a well ordered set, i.e. every nonempty subset of S has a smallest element.

L1. To obtain a contradiction, suppose that $\forall e \in S.P(e)$ is false.
L2. Let $C = \{e \in S \mid P(e) \text{ is false}\}$ be the set of counterexamples to P.
L3. $C \neq \phi$; definition: L1, L2
 L4. Let e be the smallest element of C; well ordering principle: L2, L3
  Let $e' = \cdots$
  $\vdots$
  L5. $e' \in C$
  $\vdots$
  L6. $e' < e$.
L7. This is a contradiction: L4, L5, L6
$\forall e \in S.P(e)$; proof by contradiction: L1, L7