Logic

2.1. Propositional logic

Proposition = statement that has a truth value (true or false)

ex: $\underbrace{\text{Delft}}_{\text{subject}}$ $\underbrace{\text{is a city}}_{\text{predicate}}$.

Lowercase letters ($p, q, r \ldots$) => propositional variables
=> represent a proposition

Uppercase letters ($P, Q, R \ldots$) => compound proposition

Logical operators/connectives: „$\wedge$" - and - conjunction
„$\vee$" - or - disjunction
„$\neg$" - not - negation

value	p	q	$\wedge$	$\vee$	$\neg$
	true	true	true	true	false
	true	false	false	true	true
	false	true	false	true	

A proposition made up of simpler prop. => compound proposition

When several operators of equal precedence occur in the absence of parantheses, they are evaluated from left to right.

ex: $p \wedge q \wedge r \Rightarrow (p \wedge q) \wedge r$

„$\wedge$" => associative operation

p	q	r	$p \wedge q$	$q \wedge r$	$(p \wedge q) \wedge r$	$p \wedge (q \wedge r)$
0	0	0	0	0	0	0
0	0	1	0	0	0	0
0	1	0	0	0	0	0
0	1	1	0	1	0	0
1	0	0	0	0	0	0
1	0	1	0	0	0	0
1	1	0	1	0	0	0
1	1	1	1	1	1	1

Logic

2.1. Propositional logic

Proposition = statement that has a truth value (true or false)

ex: $\underbrace{\text{Delft}}_{\text{subject}}$ $\underbrace{\text{is a city}}_{\text{predicate}}$.

Lowercase letters ($p, q, r \ldots$) => propositional variables
=> represent a proposition

Uppercase letters ($P, Q, R \ldots$) => compound proposition

Logical operators/connectives: "$\wedge$" - and - conjunction
"$\vee$" - or - disjunction
"$\neg$" - not - negation

value	p	q	$\wedge$	$\vee$	$\neg$
	true	true	true	true	p=true => $\neg p$ = false
	true	false	false	true	p= false => $\neg p$ = true
	false	true	false	true	

A proposition made up of simpler prop. => compound proposition

When several operators of equal precedence occur in the absence of paranthesses, they are evaluated from left to right.

ex: $p \wedge q \wedge r \Rightarrow (p \wedge q) \wedge r$

"$\wedge$" => associative operation

p	q	r	$p \wedge q$	$q \wedge r$	$(p \wedge q) \wedge r$	$p \wedge (q \wedge r)$
0	0	0	0	0	0	0
0	0	1	0	0	0	0
0	1	0	0	0	0	0
0	1	1	0	1	0	0
1	0	0	0	0	0	0
1	0	1	0	0	0	0
1	1	0	1	0	0	0
1	1	1	1	1	1	1

Main connective = the connective that is evaluated last

Truth table = table that shows the value of one or more compound propositions, for each possible combination (situation)

1. „¬" 2. „∧" 3. „∨"

precedence rule (in the absence of parantheses)

n variables => 2^n situations

example:

p	q	r	¬	(	p	∨	¬	q	)	∨	(	¬	r	∨	p	)
0	0	0	1	0	0	1	1	0	1	1	1	0	1	0	1	0
0	0	1	1	0	0	1	1	0	1	1	0	1	1	0	0	0
0	1	0	1	0	0	0	1	1	1	1	1	0	1	1	0	0
0	1	1	1	0	0	0	1	1	1	1	0	1	1	0	0	0
1	0	0	0	1	1	1	1	0	1	1	1	0	1	1	1	1
1	0	1	0	1	1	1	1	0	1	1	0	1	1	1	1	1
1	1	0	1	1	0	1	0	1	1	1	1	0	1	1	1	1
1	1	1	1	1	0	1	0	1	1	1	0	1	1	1	1	1

↳ the final answer

Equivalence in Propositional Logic:

1) $\neg p \vee q \overset{?}{\equiv} p \to q$

p	q	¬p	∨	q	p	→	q
0	0	1	1	0	0	1	0
0	1	1	1	1	0	1	1
1	0	0	0	0	1	0	0
1	1	0	1	1	1	1	1

They are identical, so $\neg p \vee q \equiv p \to q$

2) $(p \to q) \vee p \overset{?}{\equiv} (p \leftrightarrow q) \vee (p \leftrightarrow \neg q) \Rightarrow (p \to q) \vee p \equiv (p \leftrightarrow q) \vee (p \to \neg q)$

p	q	(p	→	q)	∨	p	(p	↔	q)	∨	(p	↔	¬q)
0	0	0	1	0	1	0	0	1	0	1	0	0	1
0	1	0	1	1	1	0	0	0	1	1	0	1	0
1	0	1	0	0	1	1	1	0	0	1	1	1	0
1	1	1	0	1	1	1	1	1	1	1	1	0	1

3) $\neg(p \vee q) \overset{?}{\equiv} \neg p \vee \neg q$

p	q	$\neg(p \vee q)$				$\neg p$	$\vee$	$\neg q$
0	0	1	0	0	0	1	1	1
0	1	0	0	1	1	1	1	0
1	0	0	1	1	0	0	1	1
1	1	0	1	1	1	0	0	0

Not equivalent!

For example: $p=0 \wedge q=1$

Then $\neg(p \vee q)$ is false

and $\neg p \vee \neg q$ is true

Logically equivalent = have the same value

Logical operators:

"$\rightarrow$" - conditional / implication

"$\leftrightarrow$" - biconditional ("$\equiv$"; "$\Leftrightarrow$")

"$\oplus$" - exclusive or

p	q	$p \rightarrow q$	$p \leftrightarrow q$	$p \oplus q$
0	0	1	1	0
0	1	1	0	1
1	0	0	0	1
1	1	1	1	0

$\underbrace{1.\ \text{"}\oplus\text{" or "}\vee\text{"} \quad 2.\ \text{"}\rightarrow\text{"} \quad 3.\ \text{"}\leftrightarrow\text{"}}_{\text{precedence rule}}$

- "$p \rightarrow q$" $\Rightarrow$ p is sufficient for q / q is necessary for p / if p then q

  p → hypothesis, q → conclusion

  $\underbrace{\text{If } p}_{\text{false}}, \underbrace{\text{then } q}_{\text{false}} \Rightarrow$ true

- "$\neg q \rightarrow \neg p$" $\Rightarrow$ contrapositive of $p \rightarrow q$

  $\Rightarrow$ the same values as $p \rightarrow q$

- $p \rightarrow q$ is not equivalent to $q \rightarrow p$ (the converse)

  $p \rightarrow q$: If this is Tuesday, then we are in Belgium

  $q \rightarrow p$: If we are in Belgium, then this is Tuesday

- $p \to q$ is <u>NOT</u> equivalent to $\neg p \to \neg q$ (the inverse)
- $p \leftrightarrow q$ is equivalent to $(p \to q) \wedge (q \to p)$

  "p if only if q" : "SK ~~will are go~~ bring u toys if and only u are good."

  " if... then": "If you are good, then SK will bring u toys."
- $p \oplus q \Rightarrow$ p or q, NOT both

  $p \vee q \Rightarrow$ p or q, or both

Equivalent propositions: $p \to q \equiv \neg p \vee q$

$p \leftrightarrow q \equiv (\neg p \vee q) \wedge (\neg q \vee p)$

$p \wedge q \equiv \neg(\neg p \vee \neg q)$

Tautology = compound proposition that is true for all possible combinations

Contradiction = c. prop. that is false for all possible combinations

Contingency = is neither a tautology nor a contradiction

$P \leftrightarrow Q$ tautology $\Rightarrow$ P and Q are equivalent

$P \equiv Q$: $(p \to q) \equiv (\neg p \vee q)$

Principle of Explosion: If I don't say the truth, then I lie

2.2 Boolean Algebra

$\mathbb{T} = 1 =$ true

$\mathbb{F} = 0 =$ false

- Double negation

  $\neg(\neg p) \equiv p$

- Excluded middle

  $p \vee \neg p \equiv \mathbb{T}$

- Contradiction

  $p \wedge \neg p \equiv \mathbb{F}$

- Identity laws

  $\mathbb{T} \wedge p \equiv p$

  $\mathbb{F} \vee p \equiv p$

- Idempotent laws

  $p \wedge p \equiv p$

  $p \vee p \equiv p$

- Commutative laws

  $p \wedge q \equiv q \wedge p$

  $p \vee q \equiv q \vee p$

- Associative laws

  $(p \wedge q) \wedge r \equiv p \wedge (q \wedge r)$

  $(p \vee q) \vee r \equiv p \vee (q \vee r)$

- Distributive laws

  $p \wedge (q \vee r) \equiv (p \wedge q) \vee (p \wedge r)$

  $p \vee (q \wedge r) \equiv (p \vee q) \wedge (p \vee r)$

- DeMorgan's laws

  $\neg(p \wedge q) \equiv (\neg p) \vee (\neg q)$

  $\neg(p \vee q) \equiv (\neg p) \wedge (\neg q)$

T2.1. First Substitution Law:

Q - any proposition and p - propositional variable & a tautology

Q is substituted for p in all places where p occurs in the tautology $\Rightarrow$ the result is also a tautology

ex: $\neg(\neg(r \to q)) \equiv r \to q$

$p \equiv r \to q$

$\neg(\neg p) \equiv p$

T2.2. 2nd substitution law

P & $Q \Rightarrow P \equiv Q$; R - compound prop. so that P is a subprop.

R' - prop where P is substituted for Q $\Rightarrow R \equiv R'$

(not necessarily to substitute every P for Q)

ex: 1) $p \wedge (p \to q) \equiv p \wedge (\neg p \vee q)$

$$\equiv (p \wedge \neg p) \vee (p \wedge q)$$

$$\equiv F \vee (p \wedge q)$$

$$\equiv (p \wedge q)$$

2) P is a tautology $\Rightarrow P \equiv T$

How to demonstrate that a compound prop. is a tautology?

$$((p \vee q) \wedge \neg p) \to q \equiv (\neg((p \vee q) \wedge \neg p)) \vee q$$

$$\equiv (\neg(p \vee q) \vee \neg(\neg p)) \vee q$$

$$\equiv (\neg(p \vee q) \vee p) \vee q$$

$$\equiv (\neg(p \vee q)) \vee (p \vee q)$$

$$\equiv T$$

- Other rules for Boolean algebra:

1) $p \vee F \equiv F \vee p$
$\equiv p$

2) $((p \vee q) \vee r) \vee s \equiv (p \vee q) \vee (r \vee s)$
$\equiv p \vee (q \vee (r \vee s))$

3) $\neg(p \to q) \equiv \neg((\neg p) \vee q)$

$$\equiv (\neg(\neg p)) \wedge (\neg q)$$

$$\equiv p \wedge \neg q$$

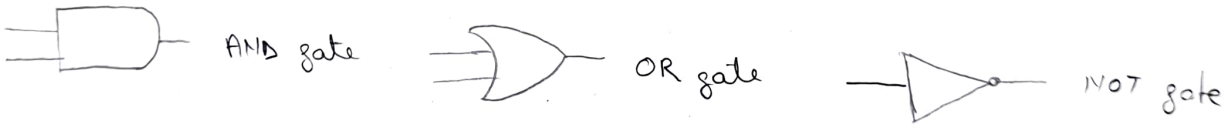
2.3. Logic Circuits

Logic gates = building blocks of computers which are electronic comp. that compute the values of simple propositions

AND gate = electronic component with two input wires and one output wire

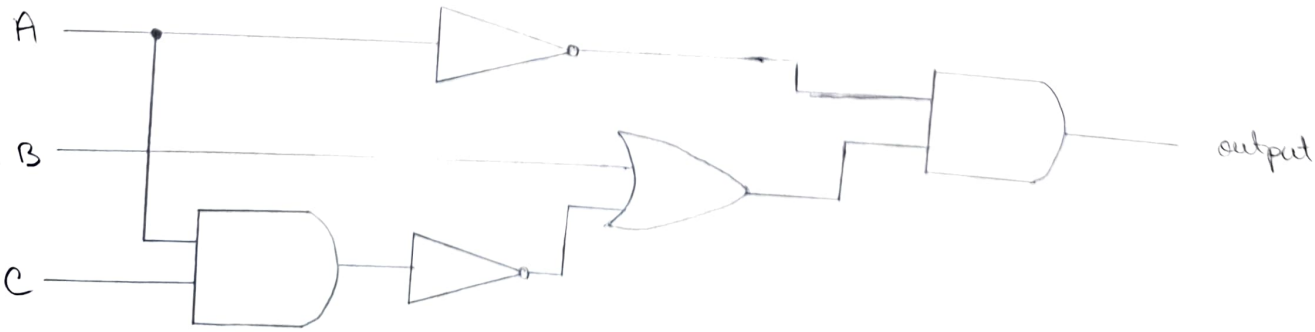
OR gate = electronic component with two inputs and one wire output which turns its output on (T) if either (or both) of its inputs are on

NOT gate = has one input and one output which turns its output off (F) when the input is on (and viceversa).



Logic circuit = logic gates wired together by connecting outputs from some gates to inputs of other gates. The gates are carrying on/off signals (T/F)

ex: $(\neg A) \wedge (B \vee \neg(A \wedge C))$



Feedback loop = occurs when the output from a gate is connected back to an input of the same gate. They can't be described by compound propositions

Combinatorial logic circuit = logic circuit that doesn't contain any feedback loops

Simple term = either a propositional variable or the negation of a propositional variable

Conjunction of simple terms = one or more simple terms put together with $\wedge$ operators

Disjunction of conjunctions of simple terms = one or more conjunctions

Disjunctive normal form (DNF):

- disj. of conj. of simple terms
- each propositional variable occurs at most once in each conjunction
- each conjunction occurs at most once in a disjunction

ex: p, q, r, s, A and B

$(p \wedge q \wedge r) \vee (p \wedge \neg q \wedge r \wedge s) \vee (\neg p \wedge \neg q)$

$(p \wedge \neg q)$

$(A \wedge \neg B) \vee (\neg A \wedge B)$

$p \vee (\neg p \wedge q) \vee (\neg p \wedge \neg q \wedge r) \vee (\neg p \wedge \neg q \wedge \neg r \wedge w)$

T. 2.3 Consider a table that lists a logical output value for every combination of values of several propositional variables. Assume that at least one of the output values is true. Then there is a proposition containing those variables such that the value of the

of the proposition for each possible combinations of the value of the variables is precisely the value specified in the table. It is possible to choose the prop. to be in disjunctive normal form.

ex: An input/output table specifying a desired output for each combination of values of the propositional variables p, q and r. Each row where the output is T corresponds to a conjunction shown next to that row in the table. The disjunction of these conj. is a proposition whose output values are precisely those specified by the table.

$(\neg p \wedge \neg q \wedge r) \vee (\neg p \wedge q \wedge r) \vee (p \wedge q \wedge r)$

p	q	r	output	
F	F	F	F	
F	F	T	T	$\Longrightarrow (\neg p \wedge \neg q \wedge r)$ conjunction
F	T	F	F	
F	T	T	T	$\Longrightarrow (\neg p \wedge q \wedge r)$ conjunction
T	F	F	F	
T	F	T	F	
T	T	F	F	
T	T	T	T	$\Longrightarrow (p \wedge q \wedge r)$ conjunction

$(\neg p \wedge \neg q \wedge r) \vee (\neg p \wedge q \wedge r) \vee (p \wedge q \wedge r) \equiv$

$\equiv (\neg p \wedge \neg q \wedge r) \vee ((\neg p \vee p) \wedge (q \wedge r))$ factor $(q \wedge r)$

$\equiv (\neg p \wedge \neg q \wedge r) \vee (q \wedge r)$ ($\neg p \vee p \equiv T$ and $T \wedge Q \equiv Q$)

$\equiv ((\neg p \wedge \neg q) \vee q) \wedge r)$ factor r

Binary addition:

$0+0+0=00$

$0+0+1=01$

$0+1+0=01$

$0+1+1=10$

$1+0+0=01$

$1+0+1=10$

$1+1+0=10$

$1+1+1=11$

then result of the sum is written in base 2.

A	B	C	output
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	1

A	B	C	output
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

In the table: there is an input / output table for each of the two outputs.

To add multi-digit binary numbers => we need one copy of the basic addition circuit for each column in the sum.

Conjunctive normal form (CNF) = conjunction of disjunctions of simple terms

2.4. Predicate Logic

One-place predicate = associates a prop. with each entity in some collection of entities

Domain of discourse = a collection of entities

- $P(a)$ = result of applying „P" to „a"

  P = predicate

  a = an entity

ex: „the rose is red" => P = „is red"
=> a = „the rose" | => $P(a)$ = „the rose is red"

Two-place predicate = associates a prop with two entities, each entity having its own domain of discourse (or not).

- $Q(a,b)$ = result of applying „Q" to „a" and „b"

  a, b = entities

  Q = predicate

ex: John loves Mary => Q = loves
John = a
Mary = b | => entities | => $Q(a,b)$

Examples of:
- three-place predicate: „a gave b to c"
- four-place predicate: „a bought b from c for d euros"

  b = must be inanimate object to make sens

ex: R = „is red" L = „loves" (two-place p)

a, b, m, j = entities

$R(a) \wedge R(b)$ = a is red and b is red

$\neg R(a)$ = a is not red

$L(j,m) \wedge \neg L(m,j)$ = j loves m and m doesn't love j

$L(j,m) \rightarrow L(b,m)$ = if j loves m then b loves m

$R(a) \leftrightarrow L(j,j)$ = a is red if and only if j loves j

Quantifiers:

- all / for all / for any => „$\forall$" universal quantifier
- no
- every
- some => „$\exists$" existential quantifier

$\forall x(P(x))$ is true if and only if $P(x)$ is true for every x in the domain on discourse

$\exists x(P(x))$ is true if and only if there is at least one entity x in the domain of discourse for P for which $P(x)$ is true

ex: $R(x)$ = „the rose is red"

$\forall x(R(x))$ = „All roses are red" (false in a way)

$\exists x(R(x))$ = „~~Some~~ rose is red" / „There is a red rose"

x = variable

Open statement = expression that contains one/more entity variables, which becomes a proposition when entities are substituted for the variables (has open „slots" that need to be filled)

Free variables = variables in an open statement

Bound variable = open statement that contains x as free variable that is quantified with $\forall x$ or $\exists x$

- $\forall y(L(x,y))$ is an open statement because x is a free variable
- $\forall x(\forall y(L(x,y)))$ or $\exists x(\forall y(L(x,y)))$ ~~is~~ a proposition
- $\exists x(\forall y(L(x,y)))$ and $\forall y(\exists x(L(x,y)))$ <u>don't</u> mean the same thing:

  $L(x,y)$ = x loves y

  $\forall y(L(x,y))$ = x loves everyone

  $\exists x(\forall y(L(x,y)))$ = There is someone who loves everyone

  $\exists x(L(x,y))$ = There is someone who loves y

  $\forall y(\exists x(L(x,y)))$ = For every person, there is somn. who loves that person

Operators with Boolean algebra:

$\underline{P(a) \to Q(a)} \equiv \neg P(a) \vee Q(a)$

↳ open statement

$\forall x(P(x) \to Q(x))$

$P(a) \wedge (\exists x\, Q(x))$

$(\forall x\, P(x)) \to (\exists x\, P(x))$

⇒ propositions

ex: $\forall x(P(x) \to Q(x))$ = All obj. with property P have property Q

$\forall x(P(x) \wedge Q(x))$ = All obj have properties P and Q

$\exists x(P(x) \to Q(x))$ = There is an obj. (such that) if it has P then it (also) has Q

$\exists x(P(x) \wedge Q(x))$ = There is an obj with P and Q

Examples of translations:

1) All flowers which are roses are red.

$\forall x (Rose(x) \rightarrow Red(x))$

2) All red roses are pretty

$\forall x ((Rose(x) \wedge Red(x)) \rightarrow Pretty(x))$

3) Jack owns a computer (There is at least one thing such that Jack owns that thing and that thing is a computer)

$\exists x (O(jack, x) \wedge C(x))$

4) Everything Jack owns is a computer.

$\forall x (O(jack, x) \rightarrow C(x))$

5) If Jack owns a computer, then he's happy.

$(\exists y (O(jack, y) \wedge C(y))) \rightarrow H(jack)$

6) Everyone who owns a computer is happy.

$\forall x ((\exists y (O(x, y) \wedge C(y)) \rightarrow H(x)))$

7) Everyone owns a computer

$\exists y \forall x (C(y) \wedge O(x, y))$

8) Everyone is happy : $\forall x H(x)$

9) Everyone is unhappy : $\forall x (\neg H(x))$

10) Someone is unhappy : $\exists x (\neg H(x))$

11) There is exactly one happy person

$\underbrace{(\exists x H(x))}_{\text{There is at least one happy person}} \wedge \underbrace{(\forall y \forall z ((H(y) \wedge H(z)) \rightarrow (y = z)))}_{\text{if } y \text{ and } z \text{ are both happy people, then they are actually the same person}}$

Logical equivalence

- DeMorgan's Laws (1)

$\neg(\forall x\, P(x)) \equiv \exists x(\neg P(x))$

$\neg(\exists x\, P(x)) \equiv \forall x(\neg P(x)$

- Other rules

$\forall x \forall y\, Q(x,y) \equiv \forall y \forall x\, Q(x,y)$

$\exists x \exists y\, Q(x,y) \equiv \exists y \exists x\, Q(x,y)$

$\neg\forall(x)(Q(x)) \equiv \exists x(\neg Q(x))$

$\neg\exists(x)(Q(x)) \equiv \forall x(\neg Q(x))$

- DeMorgan's Laws (2)

$\neg\forall y(R(y) \vee Q(y)) \equiv \exists y((\neg R(y)) \wedge (\neg Q(y)))$

P = formula of predicate logic (one/more predicate variables)

P is a tautology if it is true whenever all the predicates variables that it contains are replaced by actual predicates.

$P \equiv Q$ if $P \leftrightarrow Q$ is a tautology

2.5. Deduction

Premise = proposition that is known to be true / has been accepted to be true for the sake of argument

Conclusion = proposition that can be deduced logically from the premises

Argument = a claim that a certain conclusion follows from a given set of premises

ex: If today is Tuesday, then this is Belgium
Today is Tuesday
$\therefore$ This is Belgium

"$\therefore$" = therefore

ex: p = "Today is Tuesday" q = "This is Belgium"

$p \to q$
p
$\therefore q$

p	q	$p \to q$
0	0	1
0	1	1
1	0	0
1	1	1

ex: If Jill is breathing, then Jill pays taxes
$((p \to q) \wedge p) \to q$ is a tautology

P and Q :
- $P \to Q$ is a tautology
- in all cases where P is true, Q is also true
- $P \Rightarrow Q$ is a tautology

- P is the premise / a conjunction of premises
- „ $\Rightarrow$ " to „ $\rightarrow$ " is the same as the relation of „ $\equiv$ " to „ $\leftrightarrow$ "

Q can be logically deduced from P / P logically implies Q if $P \Rightarrow Q$, where $P \rightarrow Q$ is a tautology (all cases where P is true, Q is also true)

Valid argument = argument in which the conclusion follows logically from the premises

Modus ponens (a rule of deduction):

$$\begin{array}{l} p \rightarrow q \\ p \\ \hline \therefore q \end{array}$$

Modus tollens:

$$\begin{array}{l} p \rightarrow q \\ \neg q \\ \hline \therefore \neg p \end{array}$$

ex: If Mariah is a great person, then I'm the king of Russia
I'm not the king of Russia.

$\therefore$ Mariah is not a great person.

Law of Syllogism:

$$\begin{array}{l} p \rightarrow q \\ q \rightarrow r \\ \hline \therefore p \rightarrow r \end{array}$$

ex: If you study hard, you do well in school
If you do well in school, you get a good job

$\therefore$ If you study hard, you get a good job.

Other rules:

$$\begin{array}{l} p \vee q \\ \neg q \\ \hline \therefore q \end{array} \qquad \begin{array}{l} p \\ q \\ \hline \therefore p \wedge q \end{array} \qquad \begin{array}{l} p \wedge q \\ \hline \therefore p \end{array} \qquad \begin{array}{l} p \\ \hline \therefore p \vee q \end{array}$$

- $P \leftrightarrow Q \equiv (P \rightarrow Q) \wedge (Q \rightarrow P)$
- $P \equiv Q$ if and only if both $Q \Rightarrow P$ and $P \Rightarrow Q$

ex: $\neg(p \wedge q) \equiv (\neg p \vee \neg q)$

$\neg(p \wedge q) \Rightarrow (\neg p \vee \neg q)$

ex:

$$\begin{array}{l} (p \wedge r) \rightarrow s \\ q \rightarrow p \\ t \rightarrow r \\ q \\ t \\ \hline \therefore s \end{array}$$

ex: $P \Rightarrow Q$ and $Q \Rightarrow R$ then $P \Rightarrow R$

Formal proof: (when an arg. is valid) consists of a sequence of propositions such that the last proposition in the sequence is the conclusion of the argument (and every prop. in the seq. is either a premise of the arg. or follows by logical deduction from prop. that precede it in the list)

ex:

1.	$q \rightarrow p$	premise
2.	q	premise
3.	p	from 1 and 2 (modus ponens)
4.	$t \rightarrow r$	premise
5.	t	premise
6.	r	from 4 and 5 (modus ponens)
7.	$p \wedge r$	from 3 and 6
8.	$(p \wedge r) \rightarrow s$	premise
9.	s	from 7 and 8 (modus ponens)

- To disprove the validity of an argument, we need to provide a counterexample.
- The argument is invalid if there is even one case where all the premises are true and the conclusion is false. We need to find an assignment of truth values to prop. var. $\Rightarrow$ counterexample

ex: $p \to q$
$q \to (p \wedge r)$
r
$\therefore p$

if p is false, q is false and r true then all premises are all true but the conclusion is false

ex: $j \equiv$ John will be at the party
$m \equiv$ Mary will be there
$b \equiv$ Bill will be there
$t \equiv$ Tom will be there
$f \equiv$ The party is on Friday
$s \equiv$ The party is on Saturday

$(m \wedge \neg b) \to j$
$(f \vee s) \to m$
$b \to t$
$f \to \neg t$
f
$\therefore j$

1. $f \to \neg t$ premise
2. f premise
3. $\neg t$ from 1 and 2 (modus ponens)
4. $b \to t$ premise
5. $\neg b$ from 4 and 3 (modus tollens)
6. $f \vee s$ from 2
7. $(f \vee s) \to m$ premise
8. m from 6 and 7 (modus ponens)
9. $m \wedge \neg b$ from 8 and 5
10. $(m \wedge \neg b) \to j$ premise
11. j from 10 and 9 (modus ponens)

Deduction in predicate logic:

- if a predicate is true of all entities, then it is true of any given particular entity
- $(\forall x\, P(x)) \Rightarrow P(a)$

ex:

$$\begin{array}{l} \forall x (P(x) \rightarrow Q(x)) \\ P(a) \\ \hline \therefore Q(a) \end{array} \qquad \begin{array}{l} \forall x (P(x) \rightarrow Q(x)) \\ \neg Q(a) \\ \hline \therefore \neg P(a) \end{array}$$

Disprove the validity of an argument (counterexample):

$$\begin{array}{l} \exists x\, P(x) \\ \forall x (P(x) \rightarrow Q(x)) \\ \hline \therefore \forall x\, Q(x) \end{array} \qquad \begin{array}{l} D = \{a, b\} \\ P^{A} = \{a\} \\ Q^{A} = \{a\} \end{array}$$

- the first premis is true
- the second premise is true
- the conclusion is false