Discrete Maths: 2
Logic

SCC120 Fundamentals of Computer Science

Logic

- Logic: the study of reason, that is, of rational ways of drawing or establishing conclusions
- For this to work reliably we need a formal language (a language for reasoning)
- Types of logic: propositional logic, predicate logic, logic dealing with fuzziness, temporal logic etc.
- In this course we are concerned with propositional logic and predicate logic

Logic

- We are interested in true or false statements and how the truth/falsehood of a statement can be determined from other statements
- Examples
  - *The program runs*
  - *The program gives the right results*
  - *The program runs **and** gives the right results*

A Proposition

- Proposition: a sentence that is either true or false, but not both.
- If a proposition is true, then we say it has a truth value of "true"; if a proposition is false, its truth value is "false“

Notation	true	T, 1 (one)		false	F, 0 (zero)

Example of propositions

- "Grass is green", this proposition has the truth value of “true” (T)
- "2 + 8 = 11“, this proposition has the truth value of “false” (F)
- But!! “Turn off the computer", and "Is it hot outside ?“ are not propositions.

More examples

- The following are all propositions:
  - (a) The sentence "2+2 = 4“ happens to be a true statement, its truth value is T
  - (b)The sentence "1 = 0" is also a proposition, but its truth value is F
  - (c) "It will rain tomorrow" is a proposition. For its truth value we shall have to wait for tomorrow

More examples

- The following are examples of statements that are not propositions:
  - (a) A is less than 2
  - (b) Go for it!
  - (c) $X^2 = 11$
    - Note x does not have a fixed value! For this kind of statement see later ...

Notation

- To describe the processes of mathematical logic, we need a way of symbolizing propositions
- We use letters to represent propositions when studying logic
- Propositional Variables
  - A propositional variable can represent an arbitrary proposition

    A = “It is raining”

    B = “I feel sleepy”

New propositions from old ones

- Larger and more complex sentences are constructed from basic propositions by combining them with **connectives**
- Fundamental connectives :
  - NOT $\sim, \neg$
  - AND $\wedge$
  - OR $\vee$

Fundamental connectives

NOT	~, $\neg$
AND	$\wedge$
OR	$\vee$

- Example:
  - A = “It is raining”
  - B = “I feel sleepy”

  - A $\wedge$ B: It is raining and I feel sleepy
  - A $\vee$ B : It is raining or I feel sleepy
  - ~B: I do not feel sleepy
- The last three statements are also propositions.

Truth Table

- ($P \wedge Q$) is a proposition that takes different values depending on the values of the constituent variables P, Q
- The relationship of the value of a proposition and those of its constituent variables can be represented by a table
- This table tabulates the value of a proposition for all possible values of its variables and it is called a **truth table**

Negation (~)

- The negation of P is the statement ~P, which we read as "not p" Its truth value is defined by the following truth tables
- The negation symbol "~" is an example of a **unary logical operator** (the term "unary" indicates that the operator acts on a single element)

P	~P		P	~P
T	F		1	0
F	T		0	1

AND ($\wedge$)

- The idea of this connective is to take two propositions and make a new one which says that the original two propositions are true

P	Q	$P \wedge Q$
F	F	F
F	T	F
T	F	F
T	T	T

Ways to remember AND (∧)

P	Q	P ∧ Q
F	F	F
F	T	F
T	F	F
T	T	T

P	Q	P * Q
0	0	0
0	1	0
1	0	0
1	1	1

AND only ever returns TRUE when both/all operands are TRUE.

AND (P, Q) = TRUE if and only if P = TRUE and Q = TRUE.

Multiple Operands for AND ($\wedge$)

- We can connect as many operands as we like
  $R = A \wedge B \wedge C \wedge D$ ….
- The rule is :
  - all operands must be TRUE for the result to be TRUE
  - so, if even only one operand is FALSE, then the result is FALSE
- So the result of
  $R = T \wedge T \wedge T \wedge T \dots \wedge T \wedge F$
- is FALSE.

AND ($\wedge$) : examples

- Note: We can express $P \wedge Q$ in many different ways when using English. This tells you something about the importance of a formal logic language in computing
- - Given A, B
  - A = "Linux is an Operating System"
  - B = "Windows XP is an Operating System"
- Then
  - $A \wedge B$ = "Linux is an Operating System, and so is Windows XP"

OR (∨)

- The intent of the OR connective is to produce a statement which is true when either of its constituent parts are true (inclusive OR).

P	Q	P ∨ Q
F	F	F
F	T	T
T	F	T
T	T	T

Ways to remember OR (∨)

P	Q	P ∨ Q
F	F	F
F	T	T
T	F	T
T	T	T

P	Q	P + Q	
0	0	0	
0	1	1	
1	0	1	
1	1	1	Well, not quite!

OR only ever returns FALSE when both/all operands are FALSE.

OR (P, Q) = FALSE if and only if P = FALSE and Q = FALSE.

Multiple Operands for OR ($\vee$)

- We can connect as many operands as we like
  $R = A \vee B \vee C \vee D$ ….
- The rule is :
  - all operands must be FALSE for the result to be FALSE
  - so, if even only one operand is TRUE, then the result is TRUE
- So the result of
  $R = F \vee F \vee F \vee F \ldots \vee F \vee T$
- is TRUE.

OR ($\vee$): examples

- - Given C, D
  - C = “I am going to the Lake District”
  - D = “I am going to London”

- Then
  - $C \vee D$ = “I am going to the Lake District or I am going to London”

Complex Propositions

- By using the logical connectives given before we can construct more complex proposition
- Examples:
- **[(A ∧ B) ∨ R] ∧ [~(A ∧ B)]**

- - Given C, D
  - C = "The program runs OK"
  - D = "There is a problem with the keyword"
- Then
  - ~C ∨ D = "Either the program does not run OK or there is a problem with the keyword "

Exercise

- - Given A, B
  - A = “The program runs OK”
  - B = “There is a problem with the keyword”
- Then define the truth table of
  - $\sim A \vee B$

A	B		~A	B		$\sim A \vee B$
F	F					
F	T					
T	F					
T	T					

Tautologies

C	~C	$C \vee \sim C$
T	F	T
F	T	T

- Given proposition C:
- C = “I passed the exam”
  - ~C = “I did not pass the exam”
  - C v ~C = I passed the exam OR I did not pass the exam
- What do you think about `C v ~C`?
- The proposition has the property of always being true, regardless of the truth values of C. Statements which have this property are called **tautologies.**

Tautologies: another example

- The proposition has the property of always being true, regardless of the truth values of A and B. It is a **tautology.**
- Here we show that the expression ($A \vee B$) $\vee$ ($\sim A \wedge \sim B$) is a tautology.

A	B	~A	~B		A∨B	~A∧~B	(A∨B) ∨ (~A∧~B)
F	F	T	T		F	T	T
F	T	T	F		T	F	T
T	F	F	T		T	F	T
T	T	F	F		T	F	T

Baby Steps

A	B	~A	~B		A∨B		~A∧~B
F	F	T	T	F∨F =	F	T∧T =	T
F	T	T	F	F∨T =	T	T∧F =	F
T	F	F	T	T∨F =	T	F∧T =	F
T	T	F	F	T∨T =	T	F∧F =	F

A∨B	~A∧~B		(A∨B) ∨ (~A∧~B)
F	T	F∨T =	T
T	F	T∨F =	T
T	F	T∨F =	T
T	F	T∨F =	T

Equivalent propositions

- Two propositions are **logically equivalent** if they have exactly the same truth value under all circumstances

- Examples
  - $\sim P \vee \sim Q$ and $\sim(P \wedge Q)$ are logically equivalent
  - $\sim P \wedge \sim Q$ and $\sim(P \vee Q)$ are logically equivalent

- Whenever we find logically equivalent statement, we should feel free to substitute one for another as we wish.

Commutative properties

Associative properties

Distributive properties

De Morgan's laws

FUNDAMENTAL EQUIVALENCES

Fundamental equivalences

Commutative properties

OR	**$P \vee Q$ is equivalent to $Q \vee P$**		**$x + y = y + x$**
AND	**$P \wedge Q$ is equivalent to $Q \wedge P$**		**$x * y = y * x$**

Associative properties

OR	$(P \vee Q) \vee R$ is equivalent to $P \vee (Q \vee R)$		$(x + y) + z = x + (y + z)$
AND	$(P \wedge Q) \wedge R$ is equivalent to $P \wedge (Q \wedge R)$		$(x * y) * z = x * (y * z)$

Fundamental equivalences

Distinctive properties

$P \wedge (Q \vee R)$ is equivalent to $(P \wedge Q) \vee (P \wedge R)$	**$x * (y + z) = (x * y) + (x * z)$**
$P \vee (Q \wedge R)$ is equivalent to $(P \vee Q) \wedge (P \vee R)$	

De Morgan's Laws

$\sim P \vee \sim Q$ is equivalent to $\sim(P \wedge Q)$

$\sim P \wedge \sim Q$ is equivalent to $\sim(P \vee Q)$

Using Fundamental equivalences

- Let's look at one of DeMorgan's Laws and use it on our second example of tautologies.

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

- According to DeMorgan's second law

  $\sim P \wedge \sim Q$ is equivalent to $\sim (P \vee Q)$

- So we can change $(\sim A \wedge \sim B)$ into $\sim (A \vee B)$
- This gives us

  $(A \vee B) \vee \sim (A \vee B)$

Using Fundamental equivalences

$(A \lor B) \lor \sim(A \lor B)$

- By inspection we can see that both bracketed expressions are the same : $(A \lor B)$
- Let's say

  $C = (A \lor B)$
- Then we have $C \lor \sim C$
- Which is the same as our first example of tautology.
- We can use the fundamental equivalences to manipulate propositions with the aim of simplifying them.

Exclusive OR (XOR operator)

- The OR operator returns true as long as at least one of its operands is true. This includes the situation when both of its operands are true.
- If we need an operator that reflects the normal meaning of “or” in English ...
- Consider the proposition “I will have the soup or the salad for starters”. The OR operator returns true even if you have both soup and the salad. We need a logical operator that captures the “either .. or” nature of the way we use “or” in real life.

Exclusive OR (XOR Operator)

Inclusive OR (OR)		
P	Q	$P \vee Q$
F	F	F
F	T	T
T	F	T
T	T	T

Exclusive OR (XOR)		
P	Q	$P \underline{\vee} Q$
F	F	F
F	T	T
T	F	T
T	T	F

- XOR returns true if and only if exactly one of its operands is true. This is closer to the meaning of “or” in natural language, where we have a choice of 2 outcomes but never a choice of both outcomes together. It is either one or the other.

Conditional

- CONDITIONALS are compound propositions which take the form of "if/then" statements.
- An example is: If the train is late, then we will miss our flight.
  - The claim before the "then" but after the "if" is the antecedent of the conditional.
  - The claim after the "then" is called the consequent.
- IF antecedent THEN consequent
- IF the train is late, THEN we will miss our flight.

Conditional

- A conditional does not state either that its antecedent or that its consequent is true; rather, it states that if the antecedent is true, then the consequent is true.
- Therefore, logicians define conditionals this way: **A conditional is false if and only if its antecedent is true and its consequent is false; otherwise, a conditional is true.**

- Here is a conditional proposition.

  IF the train is late THEN we will miss our flight.

- If the train is late, but we do not miss our flight, then the above conditional must be false.

Conditional

- Let p and q be propositions. The conditional statement $p \rightarrow q$, is the proposition "if p, then q."
- The conditional statement is false when p is true and q is false, and true otherwise.

P	Q	P → Q
F	F	T
F	T	T
T	F	F
T	T	T

Conditional

- In the conditional statement $p \rightarrow q$,
  - p is called the hypothesis (or antecedent or premise) and
  - q is called the conclusion (or consequent).

- When the consequent is TRUE, the implication is also TRUE.

P	Q	P → Q
F	F	T
F	T	T
T	F	F
T	T	T
antecedent	consequent	conditional

Biconditional : $P \leftrightarrow Q$

- This table shows that connective ($P \leftrightarrow Q$) is true when both P and Q have the same truth value, and that it is false if P and Q have different truth values.

P	Q	$P \leftrightarrow Q$
F	F	T
F	T	F
T	F	F
T	T	T

Biconditional

P = “If I pass the exam”
Q = “My father will give me a gift”

$P \leftrightarrow Q$

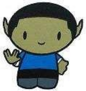
My father told me that:

He will give me a gift
if and only if I pass the exam

P	Q	$P \leftrightarrow Q$
F	F	T
F	T	F
T	F	F
T	T	T

If I do not pass the exam and my father gives me a gift, was my father telling the truth?

He was lying!
(Odd. Vulcans never lie. I don’t get on well with my dad anyway …)



Biconditional : $P \leftrightarrow Q$

- The biconditional receives its name from the fact that
- $P \leftrightarrow Q$ is equivalent to $(P \rightarrow Q) \wedge (Q \rightarrow P)$.
- We formally write this as :

  $P \leftrightarrow Q \Leftrightarrow (P \rightarrow Q) \wedge (Q \rightarrow P)$

P	Q	$P \leftrightarrow Q$
F	F	T
F	T	F
T	F	F
T	T	T

From English to Propositions

- Just to make life difficult (actually to give us a variety of different ways to say something), the proposition $P \rightarrow Q$ can be expressed in several different ways. The following are the most common:
    - If P, then Q
    - Q if P
    - P only if Q
    - P is sufficient for Q
    - Q is necessary for P

Conditional to English

- Example: Let p be the statement "Julia learns discrete mathematics." and q the statement "Julia will find a good job." Express the statement $p \rightarrow q$ as a statement in English.
- Solution: Any of the following -
- "If Julia learns discrete mathematics, then she will find a good job."
- "Julia will find a good job when she learns discrete mathematics."
- "For Julia to get a good job, it is sufficient for her to learn discrete mathematics."

Spotting conditionals

- As soon as you see something written as “if ... then ..” you should know that you’ll need a conditional to model it in propositional logic.
- “If Julia learns discrete mathematics, then she will find a good job.”

  A = “Julia learns discrete mathematics”

  B = “she will find a good job”

  $A \rightarrow B$
- But, of course, you may have to rewrite the English into an “if .. then” format.

  i.e. the last two examples on the previous slide

- antecedent $\rightarrow$ consequent
- IF antecedent THEN consequent
- "IF Julia learns discrete mathematics, THEN she will find a good job."
- A = "Julia learns discrete mathematics"
- B = "she will find a good job"
- IF A THEN B
- A $\rightarrow$ B

- antecedent $\rightarrow$ consequent
- IF antecedent THEN consequent
- "Julia will find a good job when she learns discrete mathematics."
- In English, the consequent appears before the antecedent. Be careful! Rewrite as an IF statement.
- Don't make the mistake of transforming this into "a good job" $\rightarrow$ "learns discrete mathematics" (B $\rightarrow$ A)
- "For Julia to get a good job, it is sufficient for her to learn discrete mathematics."
- As above! Rewrite as an IF statement.

We should now have covered enough logic to let you solve ...

TRUTH TABLE PUZZLES

Worked Problem 1

- Major Colorado is distraught about the theft of his fried chicken recipe.
- There are three suspects: Mr Avarice, Ms Belcher and Ms Crafty.
- Inspector Hemlock deduces from the footprints that exactly two people are involved in the theft. Further investigations reveal the following facts:

(a) Mr Avarice was involved only if Ms Crafty was not involved

(b) Ms Crafty and Ms Belcher are sworn enemies and would never join each other in any venture.

- Who are the thieves?

- A = “Mr. Avarice was involved”
- B = “Ms. Belcher was involved”
- C = “Ms. Crafty was involved”

Encoding the facts

- (a) Mr Avarice was involved only if Ms Crafty was not involved
  - If A then not C
  - $E = A \rightarrow \sim C$
- (b) Ms Crafty and Ms Belcher are sworn enemies and would never join each other in any venture.
  - $D = \sim(B \wedge C)$
- We can now evaluate these logic expressions against all possible combinations of A, B and C.

Combinations

- We have three suspects, therefore 2 to the power 3 combinations : 8.
- In the problem, we are told there are exactly two criminals.
- We can eliminate any rows that do not contain exactly two ones (1s).
- That leaves us with 3 valid rows.

Row	A	B	C
0	0	0	0
1	0	0	1
2	0	1	0
3	0	1	1
4	1	0	0
5	1	0	1
6	1	1	0
7	1	1	1

Combinations

- We have three suspects, therefore 2 to the power 3 combinations : 8.
- In the problem, we are told there are exactly two criminals.
- We can eliminate any rows that do not contain exactly two ones (1s).
- That leaves us with 3 valid rows.

Row	A	B	C
3	0	1	1
5	1	0	1
6	1	1	0

Row	A	B	C
3	F	T	T
5	T	F	T
6	T	T	F

$$D = \sim(B \wedge C)$$

B	C	(B∧C)	~(B∧C)	D
T	T	T∧T=T	F	F
F	T	F∧T=F	T	T
T	F	T∧F=F	T	T

A	B	C
F	T	T
T	F	T
T	T	F

$E = A \rightarrow \sim C$

A	C	~C	$A \rightarrow \sim C$	E
F	T	F	$F \rightarrow F = T$	T
T	T	F	$T \rightarrow F = F$	F
T	F	T	$T \rightarrow T = T$	T

A	B	C
F	T	T
T	F	T
T	T	F

Final Answer

A	B	C
F	T	T
T	F	T
T	T	F

D	E	D ∧ E
F	T	F
T	F	F
T	T	T

- Both facts (D and E) must be true.
- The result is in the bottom row.
- The guilty parties are Avarice and Belcher.

Logical Reasoning

- Definitions
  - Logical reasoning is the process of drawing conclusions from premises using rules of inference.
  - A collection of statements implies another collection of statements if whenever all the statements in the first collection are true, then all the statements in the second collection must be true.
- Inference rules are based on identities (equivalences) and implications.

List of Identities

- The list of identities gives us a set of “tools” we can use to manipulate/simplify logic expressions.
- We can replace a (sub-) expression on the LHS with its equivalent on the RHS or vice-versa

List of Identities

$(P \lor \text{True}) \Leftrightarrow \text{True}$ $(P \land \text{False}) \Leftrightarrow \text{False}$	**complement laws**
$P \Leftrightarrow (P \lor P)$	**idempotence of $\lor$**
$P \Leftrightarrow (P \land P)$	**idempotence of $\land$**
$[(P \lor Q) \lor R] \Leftrightarrow [P \lor (Q \lor R)]$ $(P \lor Q) \lor R \Leftrightarrow P \lor Q \lor R$ $P \lor Q \lor R \Leftrightarrow R \lor Q \lor P$	**associativity of $\lor$** (remove all brackets) (order does not matter)
$[(P \land Q) \land R] \Leftrightarrow [P \land (Q \land R)]$ $(P \land Q) \land R \Leftrightarrow P \land Q \land R$ $P \land Q \land R \Leftrightarrow R \land Q \land P$	**associativity of $\land$** (remove all brackets) (order does not matter)

- The above laws are used in a following example. Others are given in the appendix.

Proof of theorems

- Other laws or theorems can be defined.
- Their validity may be proved by using the laws shown before or other previously proven theorems.
- Proving laws by repeated application of previously derived laws is called **deduction**.
- Alternatively, we can prove laws using truth tables. This is called **perfect induction** (it can be tedious but often easier that deduction).

Example: Prove $(A \vee True) \Leftrightarrow True$

- Proof (using deduction).
- We are going to use a complement law : $P \vee \sim P = True$.
- We are going to use this law to convert $(A \vee True)$ into True.

P	~P	P v ~P
false	true	true
true	false	true

“Logic Ladder”

- We have reached our goal in a sequence of steps, applying identities at each turn and rewriting a statement according to these rules.

A	$\vee$	TRUE	complement law
A	$\vee$	$(A \vee {\sim}A)$	associative law
$(A \vee A)$	$\vee$	${\sim}A$	idempotence
A	$\vee$	${\sim}A$	complement law
	TRUE		

Why would we want to use these transformations?

- We can simplify (or reduce) a logical expression.
- This can save time at run-time, if the expression appears in a program.
- Can make the building of a logic circuit simpler (use less components and connections) and therefore faster.

Rule form

- Rules of inference are usually given in the following standard form:
- Premise#1
  Premise#2
  …
  <u>Premise#n</u>
  Conclusion

- In a simple case, one may use logical formulae, such as in:
- $$\begin{array}{l} P \rightarrow Q \\ P \\ \hline \quad \therefore Q \end{array}$$

List of Implications

$P \Rightarrow (P \vee Q)$	addition
$(P \wedge Q) \Rightarrow P$	simplification
$(Q \wedge P) \Rightarrow Q$	
$[P \wedge (P \rightarrow Q)] \Rightarrow Q$	modus ponens
$[(P \rightarrow Q) \wedge \sim Q] \Rightarrow \sim P$	modus tollens

List of Implications

$[(P \rightarrow Q) \wedge (Q \rightarrow R)] \Rightarrow (P \rightarrow R)$	hypothetical syllogism
$[(P \vee Q) \wedge \sim P\,] \Rightarrow Q$	disjunctive syllogism
$[(P \leftrightarrow Q) \wedge (Q \leftrightarrow R)] \Rightarrow (P \leftrightarrow R)$	

Modus Ponens: "the mode of affirming"

$$[P \wedge (P \rightarrow Q)] \Rightarrow Q$$

P	**hypothesis or premises**
(P $\rightarrow$Q)	
$\therefore$ Q	**conclusion**

The first two lines are premises or hypothesis. The last line is the conclusion.

$\therefore$ = therefore

Hypothetical syllogism

- $[(P \rightarrow Q) \wedge (Q \rightarrow R)] \Rightarrow (P \rightarrow R)$

$(P \rightarrow Q)$	If P, then Q
$(Q \rightarrow R)$	If Q, then R
$\therefore\ (P \rightarrow R)$	$\therefore$ If P, then R.

$(P \rightarrow Q)$	If it is raining, then it is cloudy
$(Q \rightarrow R)$	If it is cloudy, then I'll be sad.
$\therefore\ (P \rightarrow R)$	If it is raining, then I'll be sad.

Hypothetical Syllogism

- The hypothetical syllogism is a valid argument of the following form:
- $P \rightarrow Q$, $Q \rightarrow R$. Therefore, $P \rightarrow R$.
- It is like a transitive relationship.
- If P leads to Q, and Q leads to R, then P leads to R.

Example 1

- P= "I get up late"
- Q = "I miss the bus"
- R = "I miss my class"
- We have $(P \rightarrow Q) \wedge (Q \rightarrow R)$
- Then using hypothetical syllogism
- $[(P\rightarrow Q)\wedge(Q\rightarrow R)] \Rightarrow (P\rightarrow R)$
- We have
- If I get up late then, I miss my class

Example 2

- If I do not wake up, then I cannot go to work
- If I cannot go to work, then I will not get paid
- Therefore, if I do not wake up, then I will not get paid

Disjunctive syllogism

- $[(P \vee Q) \wedge \sim P\ ] \Rightarrow Q$

$(P \vee Q)$	P or Q
~P	Not P
$\therefore$ **Q**	$\therefore$ Q

$(P \vee Q)$	Either I will choose soup or I will choose salad.	It is red or it is black
~P	I will not choose soup.	It is not black.
$\therefore$ **Q**	Therefore, I will choose salad.	Therefore, it is red.

Modus Tollens: "the mode of denying"

- $[(P \rightarrow Q) \wedge {\sim}Q] \Rightarrow {\sim}P$

$(P \rightarrow Q)$	If P, then Q
${\sim}Q$	Not Q
$\therefore {\sim}P$	$\therefore$ Not P

$(P \rightarrow Q)$	If it is raining, then it is cloudy	If an intruder is detected by the alarm, the alarm goes off
${\sim}Q$	It is not cloudy	The alarm does not go off
$\therefore {\sim}P$	Therefore, it is not raining.	Therefore, no intruder is detected.

LOGICAL REASONING

Logic expression

- $\mathbf{C = {\sim}R \vee (R \wedge ({\sim}S \wedge {\sim}P))}$
- We can use the identities to "reduce" the expression
  - Make it smaller, less complex, more efficient
  - One of the "**absorption laws**" states that
    $\mathbf{A \vee ({\sim}A \wedge B) \Leftrightarrow A \vee B}$

Logic expression : So Far

- $C = \sim R \lor (R \land (\sim S \land \sim P))$
- **After applying absorption law**

  $A \lor (\sim A \land B) \Leftrightarrow A \lor B$

  we have
- $C = \sim R \lor (\sim S \land \sim P)$

Logic expression

- The expression is: $C = {\sim}R \lor \boxed{({\sim}S \land {\sim}P)}$
- We can apply another identity, one of DeMorgan's Laws.
- $\boxed{({\sim}A \land {\sim}B)} \Leftrightarrow {\sim}(A \lor B)$
- We have spotted that the "shape" of the boxed sub-expression matches the shape of the LHS of the Law.

Logic expression : So Far

- $C = \sim R \lor (R \land (\sim S \land \sim P))$
- After applying absorption law

  $A \lor (\sim A \land B) \Leftrightarrow A \lor B$

  we have
- $C = \sim R \lor (\sim S \land \sim P)$
- After applying DeMorgan's Law

  $(\sim A \land \sim B) \Leftrightarrow \sim(A \lor B)$

  we have our final expression
- $C = \sim R \lor \sim(S \lor P)$

APPENDIX : LIST OF IDENTITIES

List of Identities

$(P \lor \text{True}) \Leftrightarrow \text{True}$	complement laws
$(P \land \text{False}) \Leftrightarrow \text{False}$	
$(P \lor \text{False}) \Leftrightarrow P$	identity laws
$(P \land \text{True}) \Leftrightarrow P$	
$P \Leftrightarrow (P \lor P)$	idempotence of $\lor$
$P \Leftrightarrow (P \land P)$	idempotence of $\land$
$(P \lor Q) \Leftrightarrow (Q \lor P)$	commutativity of $\lor$
$(P \land Q) \Leftrightarrow (Q \land P)$	commutativity of $\land$
$[(P \lor Q) \lor R] \Leftrightarrow [P \lor (Q \lor R)]$	associativity of $\lor$
$[(P \land Q) \land R] \Leftrightarrow [P \land (Q \land R)]$	associativity of $\land$

List of Identities

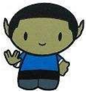
$[P \wedge (Q \vee R)] \Leftrightarrow [(P \wedge Q) \vee (P \wedge R)]$	**distributivity of $\wedge$ over $\vee$**
$[P \vee (Q \wedge R)] \Leftrightarrow [(P \vee Q) \wedge (P \vee R)]$	**distributivity of $\vee$ over $\wedge$**
$P \Leftrightarrow \sim(\sim P)$	**double negation**
$(P \rightarrow Q) \Leftrightarrow (\sim P \vee Q)$	**implication**
$(P \leftrightarrow Q) \Leftrightarrow [(P \rightarrow Q) \wedge (Q \rightarrow P)]$	**equivalence**
$(P \rightarrow Q) \Leftrightarrow (\sim Q \rightarrow \sim P)$	**contrapositive**
$\sim(P \vee Q) \Leftrightarrow (\sim P \wedge \sim Q)$	**DeMorgan's Laws**
$\sim(P \wedge Q) \Leftrightarrow (\sim P \vee \sim Q)$	**DeMorgan's Laws**

List of Identities

$P \wedge (P \vee Q) \Leftrightarrow P$	absorption laws
$P \vee (P \wedge Q) \Leftrightarrow P$	absorption laws
$P \wedge ({\sim}P \vee Q) \Leftrightarrow P \wedge Q$	absorption laws
$P \vee ({\sim}P \wedge Q) \Leftrightarrow (P \vee Q)$	absorption laws

- The list of identities gives us a set of “tool” we can use to manipulate/simplify logic expressions.
- We can replace a (sub-) expression on the LHS with its equivalent on the RHS or vice-versa.

Predicate Logic



Weakness of Propositional Logic

- Propositional logic is not powerful enough to represent all types of assertions that are used in computer science and mathematics
- For example, the assertion
  - "$x > 1$", where x is a variable, is not a proposition. Why?
  - Because you can not tell whether it is true or false unless you know the value of x
  - In this sense for x=5 "$5 > 1$" and x=-2 "$-2 > 1$" are propositions
  - Propositional logic cannot deal with statements with variables

Predicates

- Predicate: a statement which would be a proposition except for the fact that it includes variables whose values are not specified

Predicates

- Statements involving variables are neither true nor false.

  E.g. “x > 3”, “x = y + 3”, “x + y = z”
- “x is greater than 3”
  - “x”: subject of the statement
  - “is greater than 3”: the predicate
- We can denote the statement “x is greater than 3” by P(x), where P denotes the predicate and x is the variable.
- Once a value is assigned to the variable x, the statement P(x) becomes a proposition and has a truth value.

Notation

- Denote predicates by capital letters such as P,Q,R,S, etc.
- List in parenthesis the variables which are used by the predicate
- Examples (from two slides ago) :
- P(x) : $x + 4 = 3$
- S (x, y) : $x + y = 4$
- R(x) : If $x^2=4$ and $x>0$, then $x = \text{sqrt}(4)$
- T(x, y) : $x + y = 4$ if and only if $y = 4 - x$

Notation

- Note that the predicates R(x) and T(x, y) in the last slide are actually compound predicates:
  - they consist of predicates which are combined by the same kind of logical connectives used in prepositional logic
- T(x, y) : x + y = 4 if and only if y = 4 – x
  - Let's say C(x, y) : x + y = 4 , D(x, y) : y = 4 – x, two simple predicates. Then we can write T(x, y) as follows

    T(x, y) : C(x, y) $\leftrightarrow$ D (x, y)

Notation

- R(x) : If $x^2=4$ and $x>0$, then $x = \mathrm{sqrt}(4)$
- Let's say A(x) : ($x^2=4$ and $x>0$) and
  B(x) : ($x = \mathrm{sqrt}(4)$)
- Then we can write R(x) as follows
  R(x) : $A(x) \rightarrow B(x)$

Predicates

- Example: Let P(x) denote the statement "x > 3."

  What are the truth values of P(4) and P(2)?

  Solution: P(4) – "4 > 3", true

  P(2) – "2 > 3", false

Predicates

- Example: Let Q(x,y) denote the statement "x = y + 3."

  What are the truth values of the propositions Q(1,2) and Q(3,0)?

  Solution: Q(1,2) – "1 = 2 + 3" , false
  Q(3,0) – "3 = 0 + 3", true

DEFINITIONS

Definitions : Universe

- Universe: collection of values which can replace a variable in a predicate
- Examples
- $P(x) : x + 3 > 6$
- Universe for x can be the set of natural numbers
- $x \in N$

- $R(y) : ¼ + y > 4$
- Universe for y can be the set of real numbers
- $y \in R$

Definitions : n-place predicates

one-place predicate	involves one variable	R(x)
two-place predicate	involves two variables	T(x, y)
n- place predicate	involves n variables	T(x1, x2, … , xn)

Definitions : Satisfaction

- In an one-place predicate, if a value from the universe of the variable can be substituted for the variable to make the predicate true, then we say the value satisfies the predicate
- Example: $R(x) : x + 4 = 3$

  $x = -1$, so $-1 + 4 = 3$

  so, $x = -1$ satisfies $R(x)$

Definitions : n-tuple satisfaction

- If an n-tuple of values exists which can be substituted for the variables in an n-place predicate,
- and make the predicate become a true proposition,
- then we say that n-tuple satisfies the predicate

Example: n-tuple satisfaction

- $R(x, y) : x + y = 4$
- $R(x, y) = R(2,2)$, so $2 + 2 = 4$
  so, $(x, y) = (2,2)$ satisfies $R(x, y)$
- $(x, y) = (3,1)$ is another tuple that satisfies $R(x, y)$

Definitions : Satisfiable n-place predicates

- An n-place predicate is said to be satisfiable provided that there is (at least) an n-tuple which satisfies it
- Two predicates are equivalent if they have the same truth value for all possible values of their variables

QUANTIFIERS

Quantifiers

- Quantification: express the extent to which a predicate is true over a range of elements.
- Universal quantification: a predicate is true for every element under consideration
- Existential quantification: a predicate is true for one or more element under consideration
- A domain must be specified.

Quantifiers

- Universal Quantifier
- the statement $x > 1$ can be turned into "for every object x in the universe where $x > 1$", which is expressed as "$\forall x \quad x > 1$".

- Existential Quantifier
- the statement $x > 1$ can be turned into "for some object x in the universe where $x > 1$", which is expressed as "$\exists x \quad x > 1$”

Universal Quantifier

- $\forall$ x P(x) means that for all values in the universe of the variable x, the predicate P(x) is true
- In English:
  - "For all x, P(x) holds",
  - "For each x, P(x) holds" or
  - "For every x, P(x) holds"
- Example, "All cars have wheels" could be transformed into the propositional form $\forall$ x P(x) where:
  - P(x) is the predicate denoting: x has wheels, and the universe of discourse is only populated by cars

Quantification: Universal Quantifier

- Universal Quantification allows us to make a statement about a collection of objects:

"For all"

All cats are mammals

$\forall x \; \mathrm{Cat}(x) \rightarrow \mathrm{Mammal}(x)$

All of Homer's kids are also Marge's kids

$\forall x \; \mathrm{Father}(\mathrm{Homer}, x)$
$\rightarrow \mathrm{Mother}(\mathrm{Marge}, x)$

Existential Quantifier

- ∃ x P(x) means that there is at least one value in the universe of the variable x which satisfies the predicate P(x)
- In English:
  - "There exists an x such that P(x)" or
  - "There is at least one x such that P(x)”
- For example, "Someone loves you" could be transformed into the propositional form, ∃ x P(x) where:
  - P(x) is the predicate meaning: x loves you, the universe of discourse contains (but is not limited to) all living creatures

Quantification: Existential Quantifier

- Existential Quantification allows us to state that an object does exist (without naming it):

"There exists"

There is a black cat

$\exists x\ Cat(x) \wedge Black(x)$

There is a kid whose father is Homer and whose mother is Marge

$\exists x\ Father(Homer,x) \wedge Mother(Marge,x)$

Negations of Quantifier ∀

- If we claim that a universally quantified predicate is false, then we must be saying that there is some value of its variable that makes the predicate false
- `~(∀ x P(x))` is equivalent to ∃ `x ~P(x)`
- Example:

  B(x) : "the cat x is black"

  "All cats are black". For that statement to be false, it is necessary to exhibit only one cat who is not black

Negations of Quantifier ∃

- Similarly, if we claim that an existentially quantified statement is false, the only way that can occur is if the predicate is never true
- $\sim(\exists x\ P(x))$ is equivalent to $\forall x \sim P(x)$
- Example:
- B(x) : the cat x is blue
- “There is a cat who is blue”. If we claim that this is false, we can say “all cats are not blue”

TRANSLATING FROM ENGLISH INTO LOGICAL EXPRESSIONS

Coding “Not all days rain”

- P = Not all days rain (or it does not rain every day)
- For that we chose the predicate R(x) which has one argument expressing
  - R(x) : it rains during day x
- The sentence “Not all days rain” can now be coded as ~(R(x))

  Saying “it is not the case that all days are rainy”
- Alternatively, we could code this as

  $\exists x \ (\sim R(x))$

  Meaning: “There are some days x that are not rainy” or “at least there is one day that is not rainy”

Every student in this class has studied calculus

- Example: Express the above statement using predicates and quantifiers.
- Solution:
- If the domain consists of students in the class –

  $\forall x \ C(x)$

  where C(x) is the statement “x has studied calculus”.
- If the domain consists of all people –

  $$\forall x \ (S(x) \rightarrow C(x))$$

  where S(x) represents that person x is in this class.

Every student in this class has studied calculus

- From previous slide : $\forall\, x\ \ (S(x) \rightarrow C(x))$
- If we are interested in the backgrounds of people in subjects besides calculus, we can use the two-variable quantifier $Q(x, y)$ for the statement “student x has studied subject y.”
- Then we would replace $C(x)$ by $Q(x,\ calculus)$ to obtain

  $\forall\, x\ \ (S(x) \rightarrow Q(x,\ calculus))$

Consider these statements

- The first two are called premises and the third is called the conclusion. The entire set is called an argument.
  - 1. "All lions are fierce."
  - 2. "Some lions do not drink coffee."
  - 3. "Some fierce creatures do not drink coffee."
- Solution: Let P(x) be "x is a lion."

  Q(x) be "x is fierce."

  R(x) be "x drinks coffee."
- 1. $\forall x(P(x) \rightarrow Q(x))$
- 2. $\exists x(P(x) \land \neg R(x))$
- 3. $\exists x(Q(x) \land \neg R(x))$

THE END