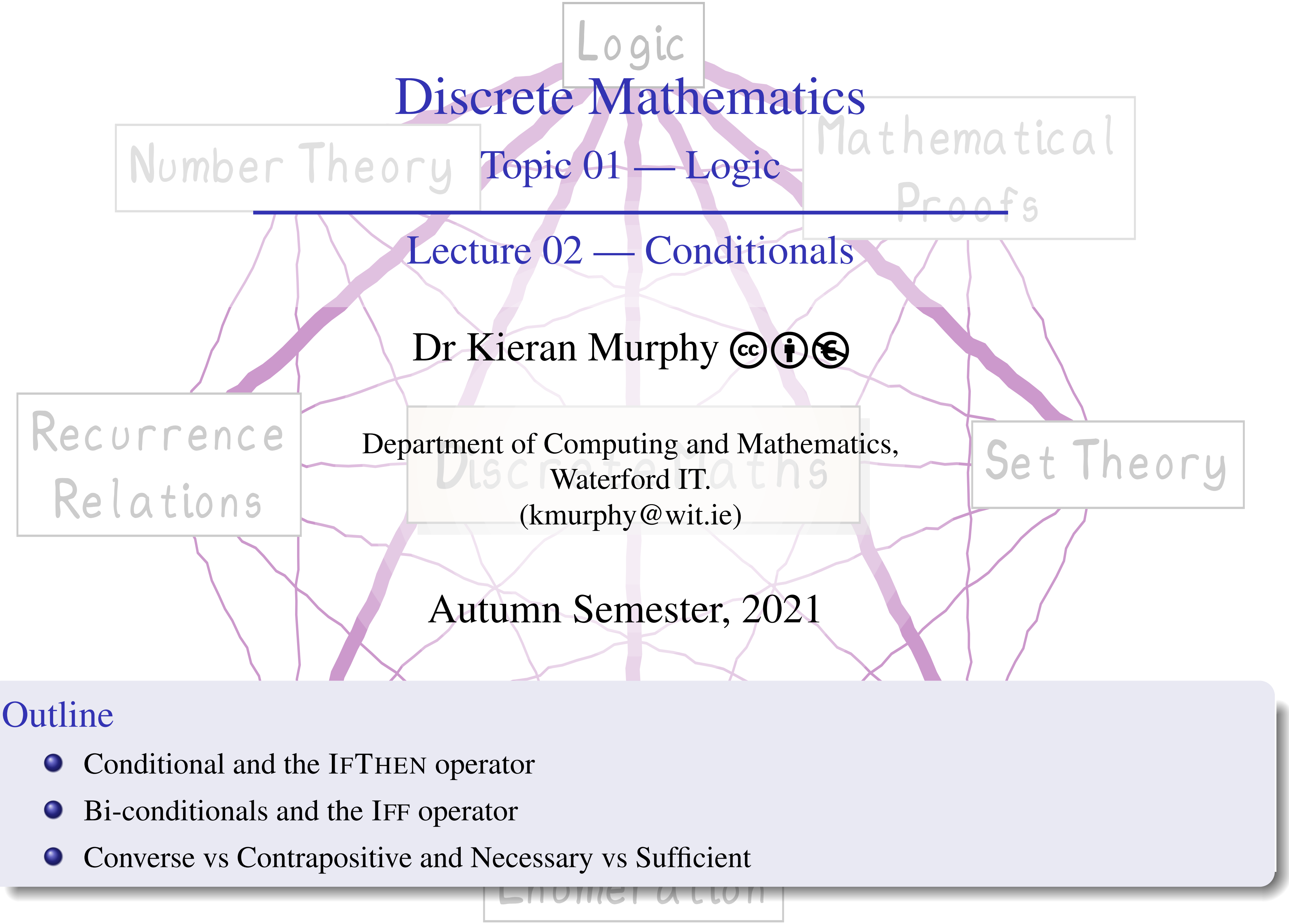

Discrete Mathematics

Topic 01 — Logic

Lecture 02 — Conditionals

Dr Kieran Murphy

Department of Computing and Mathematics,
Waterford IT.
(kmurphy@wit.ie)

Autumn Semester, 2021

Outline

- Conditional and the IFTHEN operator
- Bi-conditionals and the IFF operator
- Converse vs Contrapositive and Necessary vs Sufficient

Outline

1. Logical Equivalence 2
   - In this section we will define what we mean when we say “two logical expressions are equal” and use truth tables to test for equality.

2. Conditionals and the IFTHEN Operator 5
   - In this section we will define a logical operator, the conditional operator, that allows use to combine two logical expressions so that when ever the first expression is **True** the second expression must also be **True**.

3. Bi-conditionals and the IFF operator 12
   - In this section we will define a logical operator, the bi-conditional operator, that allows us to combine two logical expressions into a single expression that evaluates to **True** when ever both of the original expressions are equal.

4. Converse and Contrapositive 14
   - We cover with rewriting conditional expressions into its converse and contrapositive form.
   - We also look at another view of conditions — as necessary and sufficient conditions.

Logical Equivalence

Definition 1 (Logical Equivalence)

Two logical expressions are said to be logically equivalent if their truth values match whenever their inputs match.

- Given two logical expressions we can test for logical equivalence in two ways
  - Construct truth tables for each expression and compare outputs.
    - The output columns must agree for logical equivalence.
    - This is easy to do, but long, boring and error prone $\implies$ use a computer.
  - Use properties of logical operators to convert one logical expression into the other.
    - Requires knowledge of properties of logical operators (covered later).
    - Traditionally this was only done by humans, but now can be done by computers using symbolic programming.

Example

Example 2

Verify that the logical expressions $\neg(p \vee q)$ and $(\neg p) \wedge (\neg q)$ are logically equivalent.

We construct a truth table for both expressions …

p	q	$p \vee q$	$\neg(p \vee q)$
F	**F**	**F**	**T**
F	**T**	**T**	**F**
T	**F**	**T**	**F**
T	**T**	**T**	**F**

p	q	$\neg p$	$\neg q$	$(\neg p) \wedge (\neg q)$
F	**F**	**T**	**T**	**T**
F	**T**	**T**	**F**	**F**
T	**F**	**F**	**T**	**F**
T	**T**	**F**	**F**	**F**

… outputs are identical so logical expressions are equivalent.

Outline

1. Logical Equivalence 2
   - In this section we will define what we mean when we say “two logical expressions are equal” and use truth tables to test for equality.

2. Conditionals and the IfThen Operator 5
   - In this section we will define a logical operator, the conditional operator, that allows use to combine two logical expressions so that when ever the first expression is **True** the second expression must also be **True**.

3. Bi-conditionals and the Iff operator 12
   - In this section we will define a logical operator, the bi-conditional operator, that allows us to combine two logical expressions into a single expression that evaluates to **True** when ever both of the original expressions are equal.

4. Converse and Contrapositive 14
   - We cover with rewriting conditional expressions into its converse and contrapositive form.
   - We also look at another view of conditions — as necessary and sufficient conditions.

Motivation for the Conditional Operator

Consider the statement

"*If Sandra gets at least 80% in the exam, then Sandra will pass the module.*"

This is an example of an implication or a conditional, where we have two propositions

$$p = \text{“}\textit{Sandra gets at least 80\% in the exam}\text{”}$$
$$q = \text{“}\textit{Sandra will pass the module}\text{”}$$

and our statement is the claim that whenever p is **True**, then q must be **True**.

- Under what condition is this claim **False**?
- What happens if Sandra does not get at least 80%, is the claim **False**?
- We hear that Sandra has passed the module, what can we say about her mark on the exam?

Implication (IfThen) I

Definition 3 (Conditional Operator)

The conditional operator, also known as an implication or as the IfThen operator, is written as

$$p \rightarrow q$$

where propositions

- p is the hypothesis (or antecedent), and
- q is the conclusion (or consequent).

The implication

$$p \rightarrow q = \begin{cases} \textbf{False} & \text{when } p \text{ is } \textbf{True} \text{ and } q \text{ is } \textbf{False}, \\ \textbf{True} & \text{otherwise} \end{cases}$$

p	q	$p \rightarrow q$
F	**F**	**T**
F	**T**	**T**
T	**F**	**F**
T	**T**	**T**

- The implication is **True** when the hypothesis is **False** or the conclusion is **True** (or both), and **False** otherwise.
- An implication is **False** when the hypothesis is **True** and the conclusion is **False**, and **True** otherwise.

Implication (IfThen) II

In English the implication is written as

"IF *hypothesis* Then *conclusion*"

- Don't confuse the hypothesis being **True** with the conclusion being **True**, or more importantly, the implication being **True**.
- In fact, an implication is **True** whenever the hypothesis is **False**. For example, the implication

  "*If fish have hair then, chickens have lips*"

  is **True**.
- The "if" part of the sentence, the hypothesis, and the "then" part, the conclusion, of an implication need not be related in any intuitive sense. The truth or falsity of an implication is simply a fact about the logical values of its hypothesis and of its conclusion.

Examples

Decide which of the following statements are true and which are false.

1) “$0 = 1 \rightarrow 1 = 1$”

2) “$1 = 1 \rightarrow$ *most horses have 4 legs*”

3) “*If 8 is a prime number, then the 7624th digit of* π *is an 8.*”

4) “*If the 7624th digit of* π *is an 8, then* $2 + 2 = 4$.”

Example 4

Example 4

Construct a truth table for the proposition $(p \vee (\neg q)) \rightarrow ((\neg p) \wedge q)$

p	q	$\neg q$	$p \vee (\neg q)$	$\neg p$	$(\neg p) \wedge q$	$(p \vee (\neg q)) \rightarrow ((\neg p) \wedge q)$
F	**F**	**T**	**T**	**T**	**F**	**F**
F	**T**	**F**	**F**	**T**	**T**	**T**
T	**F**	**T**	**T**	**F**	**F**	**F**
T	**T**	**F**	**T**	**F**	**F**	**F**

Representing the IFTHEN in terms of other operators

Theorem 5

The implication $p \to q$ is logical equivalent to $(\neg p) \vee q$

Proof.

Construct truth tables …

p	q	$p \to q$
F	**F**	**T**
F	**T**	**T**
T	**F**	**F**
T	**T**	**T**

p	q	$\neg p$	$(\neg p) \vee q$
F	**F**	**T**	**T**
F	**T**	**T**	**T**
T	**F**	**F**	**F**
T	**T**	**F**	**T**

… outputs are identical so logical expressions are equivalent.

□

Outline

1. Logical Equivalence 2
   - In this section we will define what we mean when we say "two logical expressions are equal" and use truth tables to test for equality.

2. Conditionals and the IFTHEN Operator 5
   - In this section we will define a logical operator, the conditional operator, that allows use to combine two logical expressions so that when ever the first expression is **True** the second expression must also be **True**.

3. Bi-conditionals and the IFF operator 12
   - In this section we will define a logical operator, the bi-conditional operator, that allows us to combine two logical expressions into a single expression that evaluates to **True** when ever both of the original expressions are equal.

4. Converse and Contrapositive 14
   - We cover with rewriting conditional expressions into its converse and contrapositive form.
   - We also look at another view of conditions — as necessary and sufficient conditions.

Bi-conditional Operator

Definition 6 (Bi-conditional Operator)

The bi-conditional operator, also known as the IFANDONLYIF operator, is written as

$$p \leftrightarrow q$$

where p and q are propositions and satisfies

$$p \leftrightarrow q = \begin{cases} \mathbf{T} & \text{when } p \text{ and } q \text{ are equal} \\ \mathbf{F} & \text{otherwise} \end{cases}$$

p	q	$p \leftrightarrow q$
F	**F**	**T**
F	**T**	**F**
T	**F**	**F**
T	**T**	**T**

- The bi-conditional, $p \leftrightarrow q$ can be expressed in terms of two conditional, ie. $(p \rightarrow q) \wedge (q \rightarrow p)$.
- Two logical expressions, p and q, are said to be logically equivalent if $p \leftrightarrow q$ is **True**.
- The text “If and only if" is usually abbreviated to “iff”.

Outline

1. Logical Equivalence 2
   - In this section we will define what we mean when we say “two logical expressions are equal” and use truth tables to test for equality.

2. Conditionals and the IFTHEN Operator 5
   - In this section we will define a logical operator, the conditional operator, that allows use to combine two logical expressions so that when ever the first expression is **True** the second expression must also be **True**.

3. Bi-conditionals and the IFF operator 12
   - In this section we will define a logical operator, the bi-conditional operator, that allows us to combine two logical expressions into a single expression that evaluates to **True** when ever both of the original expressions are equal.

4. Converse and Contrapositive 14
   - We cover with rewriting conditional expressions into its converse and contrapositive form.
   - We also look at another view of conditions — as necessary and sufficient conditions.

Converse of an Implication

Definition 7 (Converse)

The converse of an implication $p \to q$ is the implication $q \to p$.

- The converse is **NOT** logically equivalent to the original implication. That is, whether the converse of an implication is true is independent of the truth of the implication.

Example

The converse of the implication

“*If x is a healthy horse then x has four legs*”

is the statement

“*If x has four legs then x is a healthy horse*”

Hopefully, you will agree that the these two statements are not logically equivalent. (What happens if x is Donald, my pet aardvark?)

Contrapositive

Definition 8 (Contrapositive)

The contrapositive of an implication $p \rightarrow q$ is the statement $(\neg q) \rightarrow (\neg p)$.

- An implication and its contrapositive are logically equivalent (they are either both **True** or both **False**.
- Hence the contrapositive always has the same truth value as its original implication.
- However, in some situations it is easier to work with the contrapositive rather than the original implication.

Example 9 II

Example 9

Suppose I tell Bob that if he gets a 93% in the exam, then he will get an A in the module. Assuming that what I said is true, what can you conclude in the following cases:

1) *“Bob gets a 93% in his exam.”*
2) *“Bob gets an A in the module.”*
3) *“Bob does not get a 93% in his exam.”*
4) *“Bob does not get an A in the module.”*

Here we have propositions

$$p = \text{“\textit{Bob gets a 93\% in his exam}”}$$
$$q = \text{“\textit{Bob gets an A in the module}”}$$

and implication

$$p \to q$$

which we are assuming is **True**.

Example 9 II

Implication $p \rightarrow q$

"If Bob get a 93% in his exam, then Bob will get an A in the module."

Converse $q \rightarrow p$

"If Bob gets an A in the module, then Bob got a 93% in his exam."

Contrapositive $(\neg q) \rightarrow (\neg p)$

"If Bob does not get an A in the module, then Bob did not got a 93% in his exam."

Contrapositive of the converse $(\neg p) \rightarrow (\neg q)$

"If Bob does not get a 93% in his exam, then Bob will not get an A in the module."

1) *"Bob gets a 93% in his exam."*

2) *"Bob gets an A in the module."*

3) *"Bob does not get a 93% in his exam."*

4) *"Bob does not get an A in the module."*

Necessary and Sufficient

We will end this topic on conditionals by introducing alternative terminology that is commonly used when designing programs or systems …

Definition 10 (Necessary and Sufficient)

- "*p is necessary for q*" means $q \to p$.
- "*p is sufficient for q*" means $p \to q$.
- "*p is necessary and sufficient for q*" means $p \leftrightarrow q$.

- The terms necessary and sufficient are often used when designing programs and specifying the conditions under when a program will operate correctly.