Mathematical Foundations for Software Engineering

Course notes

Attila Matolcsy

25th August 2023 - Present

Contents

1 About

This document is Attila Matolcsy's own personal notes from the University of Gothenburg's Software Engineering and Management Bsc. Programme's Mathematical Foundations for SEM course.

Introduction day

Lessons are not mandatory, nor are the TA lessons. We can choose our TA groups we'd like to work in.

Working on projects and assignments are also allowed in groups of up to 3 members.

On Canvas there are educational materials, course literatures (although not mandatory) and more information is available.

The teacher can be contacted through e-mail at christian.berger@gu.se.

2 Logic

Mathematical logic works with statements. Statements are declerative, meaning they are either true or false.

We have applications, in the design of computing systems we usually use 1s and 0s.

Proofs are mathematical arguments and these are essenctial for programs.
We use proofs in computer security systems as well.

Theorems are proven mathematical statements and propositions are statements that are either True (T) or False (F).
For propositions we commonly use lettering starting from p, so $p, q, r, s, \ldots$

2.1 Logical Operators

Def. 1: Negation: The opposite of the proposition ($T \to F$ & $F \to T$)
Notation: $\neg$

Def. 2: Conjuction: logical AND, the output of a conjuction is only true when the input statements are all true.
Notation: $\wedge$

p	q	$p \wedge q$
T	T	T
T	F	F
F	T	F
F	F	F

p	q	$p \wedge q$
1	1	1
1	0	0
0	1	0
0	0	0

Def. 3: Disjuction: logical OR, the output of a logical OR is true when at least one of the input statements are true.
Notation: $\vee$

p	q	$p \vee q$
T	T	T
T	F	T
F	T	T
F	F	F

p	q	$p \vee q$
1	1	1
1	0	1
0	1	1
0	0	0

Def. 4: Excl.OR: exclusive OR, the output of an exclusive OR is true when only one of the input statements are true.
Notation: $\oplus$

p	q	$p \oplus q$
T	T	F
T	F	T
F	T	T
F	F	F

p	q	$p \oplus q$
1	1	0
1	0	1
0	1	1
0	0	0

Def. 5: Implication: implication (AKA: If this then this), the output of an implication is false when the first statement is true, but the implied statement is false. In other cases it's true.
Notation: $\Rightarrow$

p	q	$p \Rightarrow q$
T	T	T
T	F	F
F	T	T
F	F	T

p	q	$p \Rightarrow q$
1	1	1
1	0	0
0	1	1
0	0	1

Def. 6: Biconditional statements: bidirectional implication, the output of a biconditional statement is true when the both of the propositions have the exact same value
Notation: $\Leftrightarrow$

p	q	$p \Leftrightarrow q$
T	T	T
T	F	F
F	T	F
F	F	T

p	q	$p \Leftrightarrow q$
1	1	1
1	0	0
0	1	0
0	0	1

Def. 7: Tautology, A tautology is when the statement is always true, regardless of the propositions' values.

$$\text{e.g.: } p \vee \neg p \equiv T \equiv 1$$

Def. 8: Contardiction, A contradiction is when the statement is always false, regardless of the propositions' values.

$$\text{e.g.: } p \wedge \neg p \equiv F \equiv 0$$

Def. 9: Contingency: It's neither tautology nor contradiction.
The output proposition is not related to the input propositions.

Def. 10: Equivalents:
Statements that are equal

$$p \wedge T \equiv p \qquad p \wedge 1 \equiv p$$
$$p \vee T \equiv T \qquad p \vee 1 \equiv 1$$
$$p \wedge F \equiv F \qquad p \wedge 0 \equiv 0$$
$$p \vee F \equiv p \qquad p \vee 0 \equiv p$$

2.2 Using binary numbers for truth tables

In truth tables we always shown all the possible variations of each statement. During this it's easier to view those as binary numbers, so we can easily check if we written down all the possible combinations.

Note that this is not a requirement that you do it this this way.
This is just a surefire method to make sure you do not miss any cases, also that all of your truth tables are similar.

When looking at the number of propositions we can learn that there are $o = 2^p$ options, where o is the number of options and p is the number of propositions.

So in case of 3 propositions, we have $2^3 = 8$ options.
These are:

p	q	r	p	q	r
T	T	T	1	1	1
T	T	F	1	1	0
T	F	T	1	0	1
T	F	F	1	0	0
F	T	T	0	1	1
F	T	F	0	1	0
F	F	T	0	0	1
F	F	F	0	0	0

...	p	q	r
x increases from right to left: 2^x This is what the binary number represents:	$2^2 = 4$	$2^1 = 2$	$2^0 = 1$
This is our binary number:	$1\,(T)$	$1\,(T)$	$1\,(T)$
We need to sum this: $4 + 2 + 1 = 7$	$1 \cdot 4 = 4$	$1 \cdot 2 = 2$	$1 \cdot 1 = 1$

We started from $1\,1\,1$ or 111_b, we always subtract 1 from our binary number, so it goes $110_b, 101_b, 100_b$, etc.

Note that binary digits can take up only 2 values: $0, 1$. So when you subtract 1 from a binary number ending with 0, it doesn't become 9, but rather 1 again. So $10_b - 1 \neq 09$, but rather $10_b - 1 = 01_b$. Also we notate the binary numbers usually with $_b$ which stands for binary or $_2$ which stands for the number of values a proposition can have (True, False / $1, 0$).

2.3 Quantifiers (Predicate logic)

We call statements that contain variables predicates.

3 Credits

Real Analysis - Foundations and Functions of One Variable – by: Miklós Laczkovich , Vera T. Sós
Real Analysis - Series, Functions of Several Variables, and Applications – by: Miklós Laczkovich , Vera T. Sós