4COSCOO2 Mathematics for Computing,
TUTORIAL 1 TASKS and ANSWER SCHEMES

Aim: introduction to Booleans, practice identifying primitive and compound propositions and simple cases of using Booleans to form compound propositions, simple truth tables.

Tasks:

1. For each of the examples of reasoning below, answer the following questions:
   - Find all Boolean expressions identifying all primitive propositions and the structure of the compound proposition.
   - Is the conclusion correct?
   - If it is correct, explain why.
   - If it is not correct, give a counter example.

   a) If X is divisible by 4 then it is divisible by 2.

   b) If X is divisible by 4 or 6 then it is divisible by 2 or 3.

   c) If X is divisible by 2 then it is divisible by 4.

   d) If X is not divisible by 2 then it is not divisible by 4.

   e) If X is not divisible by 2 and not divisible by 3 then it is not divisible by 4 and it is not divisible by 6.

2. For the following formal expressions establish if their grammar is correct or indicate a problem in their logical structure otherwise.

		ANSWERS
P	YES/NO	
$\Rightarrow p$	YES/NO	
$\neg\Rightarrow p$	YES/NO	
$p\ q$	YES/NO	
$p\wedge q$	YES/NO	
$p\neg q$	YES/NO	
$p\wedge\neg q$	YES/NO	
$(\neg p\vee q$	YES/NO	
$(\neg p\vee)q$	YES/NO	
$\neg(\neg p\wedge\neg q)$	YES/NO	
$(p\Rightarrow q)\wedge(q\Rightarrow p)$	YES/NO	
$(p\Rightarrow(q\wedge r))\Rightarrow((p\Rightarrow q)\wedge(p\Rightarrow r))$	YES/NO	

3. Build truth tables for the formulae below. You should follow the algorithm described in the lecture.

a. $\neg(p\Rightarrow p)$
b. $\neg(p\vee\neg p)$
c. $(p\wedge q)\Rightarrow p$
d. $\neg(q\vee\neg q)$
e. $p\Rightarrow(p\vee q)$
f. $(s\wedge t)\Rightarrow s$
g. $u\Rightarrow(u\vee w)$

Look at the structure of the given formulae and their truth tables – can you see some patterns here? Can you identify those expressions that have similar logical structures, compare their truth tables.

4. CHALLENGE.

a) Build a truth table for the following expression following the algorithm:

i. $\leftarrow((s \ni t)$ ® $(s\vee r))$

Can you find input values for which the resulting value is true?

b.) Consider the following two expressions:

ii. $\leftarrow\leftarrow((s \ni t)$ ® $(s\vee r))$ iii. $\leftarrow\leftarrow\leftarrow((s \ni t)$ ® $(s\vee r))$

Can you obtain the truth tables for ii. and iii. only looking at the case i.?

ANSWERS:

Answers:

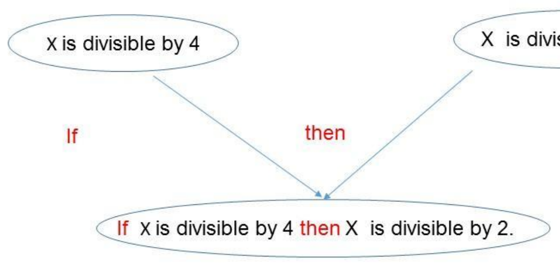
1.Consider the following examples of reasoning and answer the following questions for each of them:

- Find all Boolean expressions identifying all primitive expressions and the structure of the compound expression.
- is the conclusion correct?
- if it is correct, explain why.
- If it is not correct, give a counter-example.

a) If X is divisible by 4 then X is divisible by 2.

Booleans are coloured in red. This is a compound conditional proposition with the following primitives:

"X is divisible by 4" and "X is divisible by 2". Boolean "If …. Then" is used to make a conditional proposition, so we can graphically represent it as follows:



This is a conditional proposition and this is true: since 4 is divisible by 2, every number which is divisible by 4 is also divisible by 2

b) If X is divisible by 4 or 6 then X is divisible by 2 or 3.

This is a conditional proposition but this is more complex – since the condition on the left gives us two options – a number is divisible by 4 or 6, where "or" is a logical operation – disjunction.

We have four primitive propositions here:

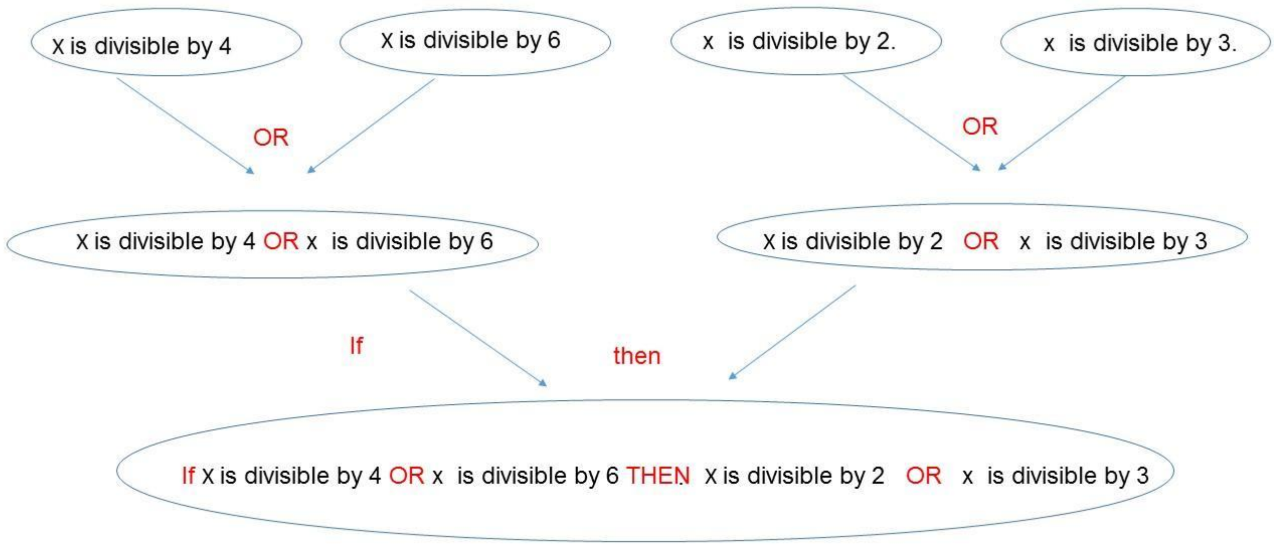
"X is divisible by 2"

"X is divisible by 3"

"X is divisible by 4"

"X is divisible by 6"

And graphically a compound expression is constructed as follows:



We also know that if a number is divisible by 4 then it is divisible by 2. Similarly, if a number is divisible by 6 then it is divisible by 3. So whichever of the conditions on the left is true, the condition on the right becomes true as one of its options becomes true.

c) If X is divisible by 2 then X is divisible by 4.

Primitives are the same as in a) and the compound structure can be explained in the same way. The compound proposition here is not correct – take, for example, number 2 itself – it is divisible by 2 but not by 4.

d) If X is not divisible by 2 then it is not divisible by 4.

Again, the primitives are the same as in a). However, on the left of the implication we have a compound proposition "X is not divisible by 2" which is obtained from a primitive "X is divisible by 2" by adding the negation to it. Similarly, we obtain "X is not divisible by 4" from the primitive "X is divisible by 4". Finally we use these two compound negative propositions to assemble an implicative proposition.

This compound proposition is correct as any number divisible by 4 should be divisible by 2. We will see later that this is a contraposition to case a).

e) If X is not divisible by 2 and not divisible by 3 then it is not divisible by 4 and it is not divisible by 6.

This is a conditional proposition but this is more complex – since the condition on the left is conjunctive – a number is not divisible by 2 and by 6, where "and" is a logical operation – conjunction. We also know that if a number is divisible by 4 then it is divisible by 2. So if X is not divisible by 2 then it should not be divisible by 4. Similarly, if a number is divisible by 6 then it is divisible by 3 so if it is not divisible by 3 it should not be divisible by 6. So if the compound condition on the left is true, then the compound condition on the right of the implication becomes true.

Task 3. For the following formal expressions establish if their grammar is correct or indicate a problem in their logical structure otherwise.

		ANSWERS
p	YES/NO	YES/ it is an atomic proposition
$\Rightarrow p$	YES/NO	NO, should be a variable/proposition on the left hand side
$\neg\Rightarrow p$	YES/NO	NO, possibly missing variable after negation
$p\ q$	YES/NO	NO, missing Boolean between variables
$p\wedge q$	YES/NO	YES, it is logical conjunction between p, q
$p\neg q$	YES/NO	NO missing variable after negation
$p\wedge\neg q$	YES/NO	YES, Remember that we agreed the negation is the strongest Boolean so we do not use brackets with it
$(\neg p\vee q$	YES/NO	NO, left bracket does not have a counterpart right one
$(\neg p\vee)q$	YES/NO	NO, MESS WITH BRACKETS
$\neg(\neg p\wedge\neg q)$	YES/NO	Yes
$(p\Rightarrow q)\wedge(q\Rightarrow p)$	YES/NO	YES
$(p\Rightarrow(q\wedge r))\Rightarrow((p\Rightarrow q)\wedge(p\Rightarrow r))$	YES/NO	YES

5. Build truth tables for the formulae below. You should follow the algorithm described in the lecture.

a. $\neg (p \Rightarrow p)$
b. $\neg (p \vee \neg p)$
c. $(p \wedge q) \Rightarrow p$
d. $\neg (q \vee \neg q)$
e. $p \Rightarrow (p \vee q)$
f. $(s \wedge t) \Rightarrow s$
g. $u \Rightarrow (u \vee w)$

Look at the structure of the given formulae and their truth tables – can you see some patterns here? Can you identify those expressions that have similar logical structures, compare their truth tables.

Answers:

From the formulae below you will see that b and c have similar structure – only symbols for atomic propositions are different! The same can be observed for the pair: d and f.

In all examples – numbers on the top of operations refer to the order in which operations are computed.

a) ← (*p* ® *p*)

Apply the algorithm to *C* which in our case is $\leftarrow (p \Rightarrow p)$

1.List its atomic propositions of C and count their number, *n = 1*.

2. Form a table with $m = 2^1 = 2$ rows.

3. List all possible combinations of the truth values for *p*: T, F

4. Prioritize the logical operations in C*: implication is first, negation - second*

2 1

5. Compute $\leftarrow (p \Rightarrow p)$

Truth table for $\overset{2}{\neg}(p \overset{1}{\Rightarrow} p)$

$\boldsymbol{p}$	$\overset{1}{p \Rightarrow p}$	$\overset{2}{\neg}(p \Rightarrow p)$
T	T	F
F	T	F

Truth table for $\overset{3}{\neg}\,(p \overset{2}{\vee} \overset{1}{\neg}\, p)$

p	$\overset{1}{\neg p}$	$\overset{2}{p \vee \neg p}$	$\overset{3}{\neg (p \vee \neg p)}$
T	F	T	F
F	T	T	F

Truth table for $\overset{3}{\neg}(q \overset{2}{\vee} \overset{1}{\neg} q)$

q	$\overset{1}{\neg q}$	$\overset{2}{q \vee \neg q}$	$\overset{3}{\neg (q \vee \neg q)}$
T	F	T	F
F	T	T	F

Truth table for $p \overset{2}{\Rightarrow} (p \overset{1}{\vee} q)$

p	q	$\overset{1}{p \vee q}$	$\overset{2}{p \Rightarrow (p \vee q)}$
T	T	T	T
T	F	T	T
F	T	T	T
F	F	F	T

Truth table for $(s \overset{1}{\wedge} t) \overset{2}{\Rightarrow} s$

s	t	$\overset{1}{s \wedge t}$	$\overset{3}{(s \wedge t) \Rightarrow s}$
T	T	T	T
T	F	F	T
F	T	F	T
F	F	F	T

Truth table for $u \overset{2}{\Rightarrow} (u \overset{1}{\vee} w)$

u	w	$\overset{1}{u \vee w}$	$\overset{2}{u \Rightarrow (u \vee w)}$
T	T	T	T
T	F	T	T
F	T	T	T
F	F	F	T

b) CHALLENGE.

Build a truth table for the following expression following the algorithm:

i. $\leftarrow((s \ni t) ® (s \vee r))$

Can you find input values for which the resulting value is true?

Consider the following two expressions:

ii. $\leftarrow\leftarrow((s \ni t) ® (s \vee r))$

iii. $\leftarrow\leftarrow\leftarrow((s \ni t) ® (s \vee r))$

Can you obtain the truth tables for ii. and iii. only looking at the case i.?

Note – this is the first example with three variables while all other examples for truth table had 1 or 2 variables. The students are aware of the algorithm to list all distributions of the values T and F from lecture 1.

ANSWER:

Apply the algorithm to *C*: $\leftarrow((s \ni t) ® (s \vee r))$

List its atomic propositions of C and count their number, $n = 3$.

2. Form a table with $m = 2^3 = 8$ rows.

3. List all possible combinations of the truth values for *p*: these will go into the left hand side of the truth table as 8 possible input values

4. Prioritize the logical operations in C*: conjunction -1, disjunction – 2, implication – 3, negation - 4*

5. Compute

Truth table for $\overset{4}{\neg}((s \overset{1}{\wedge} t) \overset{3}{\Rightarrow} (s \overset{2}{\vee} r))$

s	t	r	1 $s \wedge t$	2 $s \vee r$	3 $(s \wedge t) \Rightarrow (s \vee r)$	4 $\neg((s \wedge t) \Rightarrow (s \vee r))$
T	T	T	T	T	T	F
T	T	F	T	T	T	F
T	F	T	F	T	T	F
T	F	F	F	T	T	F
F	T	T	F	T	T	F
F	T	F	F	F	T	F
F	F	T	F	T	T	F
F	F	F	F	F	T	F

From the truth table we can see that there is no input value which makes the resulting value true.

1. Consider the following two expressions:

   ii. ←←((*s* ∍ *t*) ® (*s* V *r*))

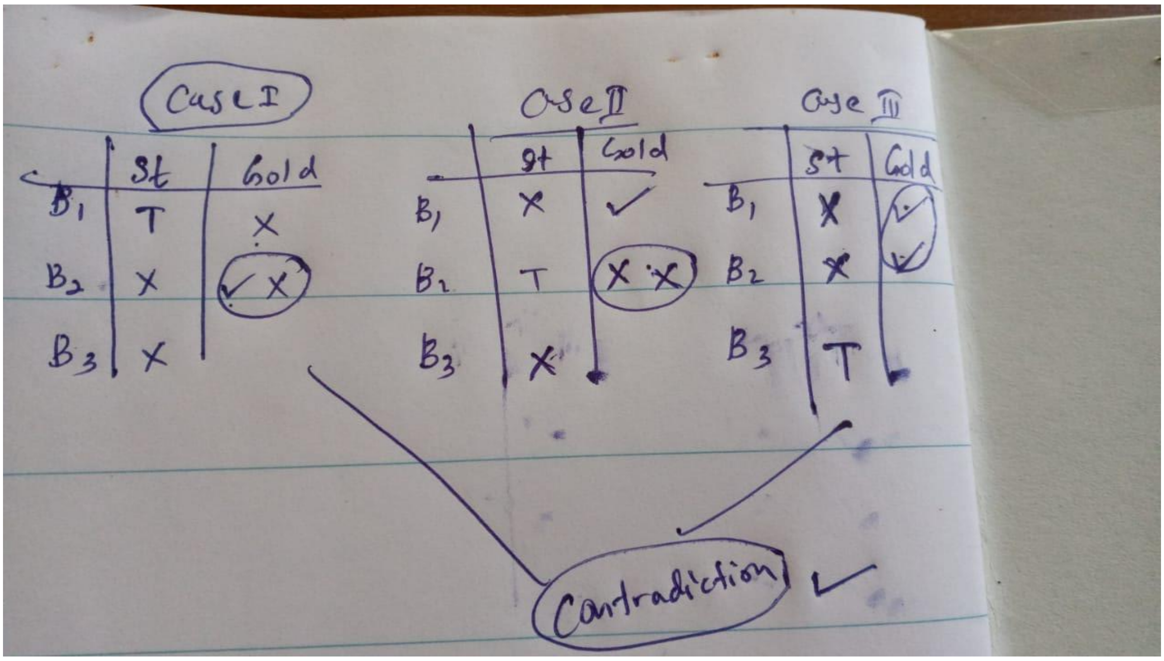
   iii. ←←←((*s* ∍ *t*) ® (*s* V *r*))

Can you obtain the truth tables for ii. and iii. only looking at the case i.?

Answer: just look at the number of negations! Formula ii. just adds one negation to i. so its truth tables will have the resulting value with all T

Formula iii. just adds one negation to ii. so its truth tables will have the resulting value with all F - as the initial formula in i.

8.



Consider three cases where only one statement is true at a time as above. Then analyse each case with the assumption made on the 3clues and find where gold is. Case I and II gives contradictions(case I: gold is in Box 2 and gold is not in Box 2, Case III: Gold is in both boxes Box 1 and Box 2) so gold is box 1 where only the clue in box 2 is true and others false.