Objectives for this lesson

3.2.2 Fundamental Hardware Elements of Computers

Logic Gates		Construct truth tables for the following gates: NOT, AND, OR, XOR, NAND, NOR.
		Be familiar with drawing logic diagrams involving one or more of the above gates.
Boolean Algebra		Be familiar with the use of De Morgan's laws and Boolean identities to manipulate and simplify simple Boolean expressions.

DE MORGAN'S LAWS

NOT (A AND B) = (NOT A) OR (NOT B)

NOT (A OR B) = (NOT A) AND (NOT B)

Augustus De Morgan (1806-1871)

- British mathematician and logician
- formulated De Morgan's laws
- introduced the term mathematical induction, making its idea rigorous

Set theoretic representation

$$(A \cup B)' = A' \cap B'$$

$$(A \cap B)' = A' \cup B'$$

- **The complement of the union of two sets is the intersection of their complements**
- **the complement of the intersection of two sets is the union of their complements.**

Example (Union Law)

1) **Let U = {1, 2, 3, 4, 5, 6}, A = {2, 3} and B = {3, 4, 5}.**

Show that $(A \cup B)' = A' \cap B'$.

Solution :

U = {1, 2, 3, 4, 5, 6}

A = {2, 3}

B = {3, 4, 5}

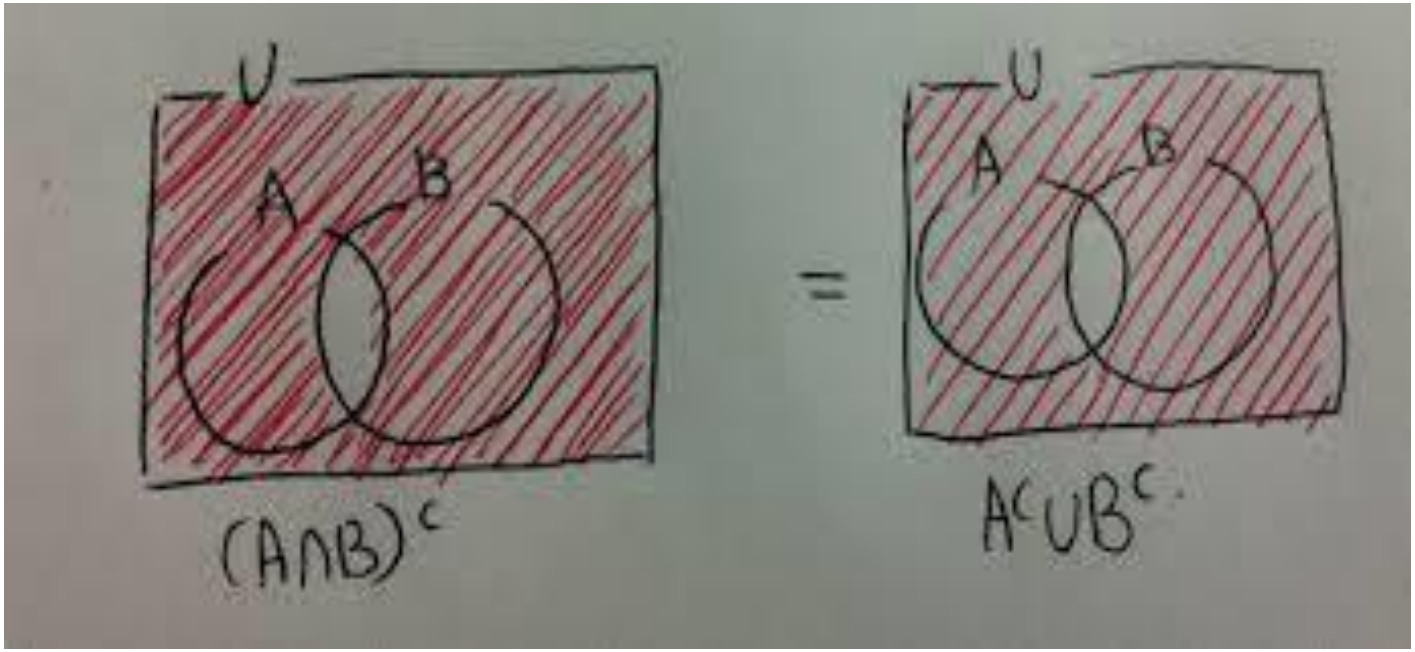
$A \cup B = \{2, 3\} \cup \{3, 4, 5\}$

$= \{2, 3, 4, 5\}$

$\therefore (A \cup B)' = \{1, 6\} = A' \cap B'$.

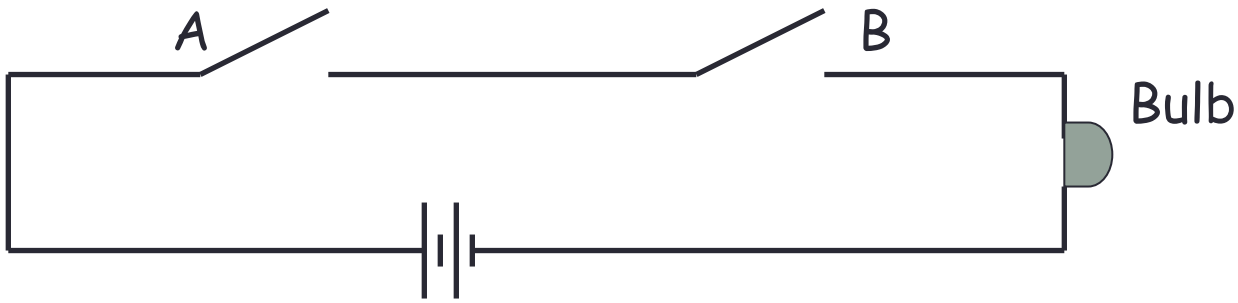
Example 2: (Intersection Law)

$(A \cap B)' = A' \cup B'$



Logic Gates

- As a first example consider a lighting circuit with two switches A and B



Q. Under what conditions will the bulb light ?

A.

The key to this problem is to realise that the bulb will light **ONLY** when both switches A AND B are closed.

This can be represented in a **Truth Table**.

In binary logic we denote a zero or low voltage by a digital 0 and a high voltage by a digital 1.

This type of table is called a

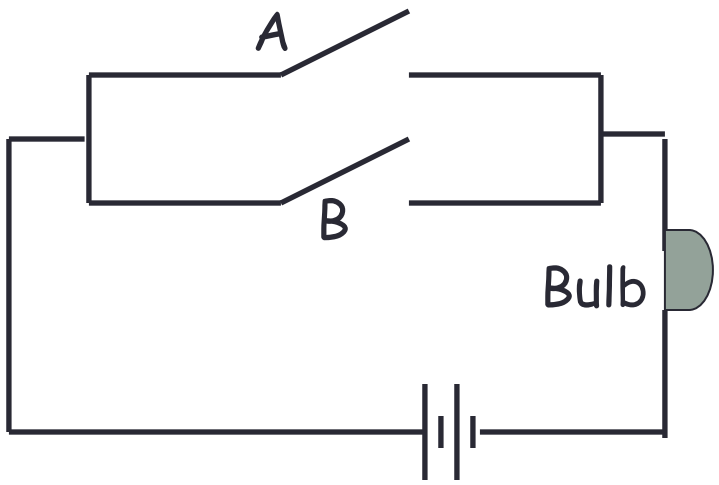
2 Input AND Truth Table with 1 output

2 Input AND Truth Table with 1 output

Switch A	Switch B	Output Y
Open	Open	NO
Open	Closed	NO
Closed	Open	NO
Closed	Closed	YES

A	B	Y
0	0	0
0	1	0
1	0	0
1	1	1

Another example



In this arrangement the bulb can light if **either** switch A or B is closed.

This is an example of a circuit based on an **OR** gate.

What is the name for the Truth Table for this arrangement?

2 input OR Table

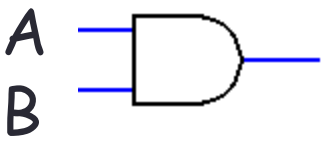
The 2 input OR table shows that
if **one or both** inputs A and B are **digital** 1, the output is a digital 1.

The Boolean expression for an OR gate is Y= A + B and is read as

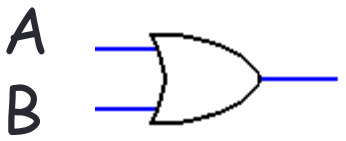
'Y is the output of A OR B'.

A	B	Y
0	0	0
0	1	1
1	0	1
1	1	1

These logic gates are represented by

AND gate 

$Y = A \bullet B$ (also $Y=AB$)

OR gate 

$Y = A+B$

Both of these gates have 2 inputs and 1 output. A larger number of inputs are possible (see later).

The NOT gate or **inverter** consists of a single input and the output is the opposite or **complement** of the input

NOT gate 

$Y = \overline{A}$ or A'

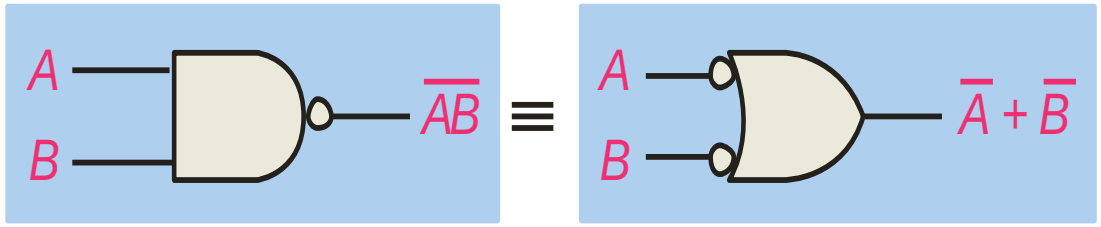
A	Y
0	1
1	0

For example $\overline{0} = 1$ and $\overline{1} = 0$

Applying De Morgan's 1st law to gates

$$\overline{AB} = \overline{A} + \overline{B}$$

Inputs		Output	
A	B	$\overline{AB}$	$\overline{A} + \overline{B}$
0	0	1	1
0	1	1	1
1	0	1	1
1	1	0	0

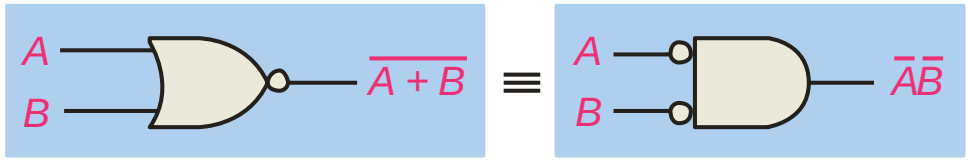


NAND Negative-OR

Similarly for the 2nd law:

$$\overline{A + B} = \overline{A} \cdot \overline{B}$$

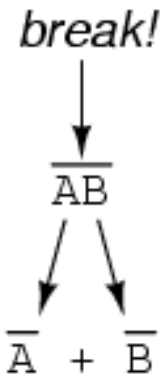
Inputs		Output	
A	B	$\overline{A + B}$	$\overline{A}\,\overline{B}$
0	0	1	1
0	1	0	0
1	0	0	0
1	1	0	0



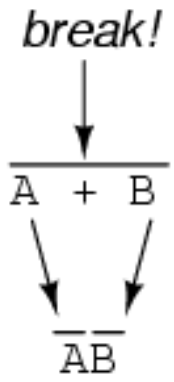
NOR Negative-AND

When a long bar is broken the operation directly underneath the break changes

- De Morgan's laws may be thought of in terms of ***breaking*** a long bar symbol.
- the broken bar pieces remain over the individual variables.

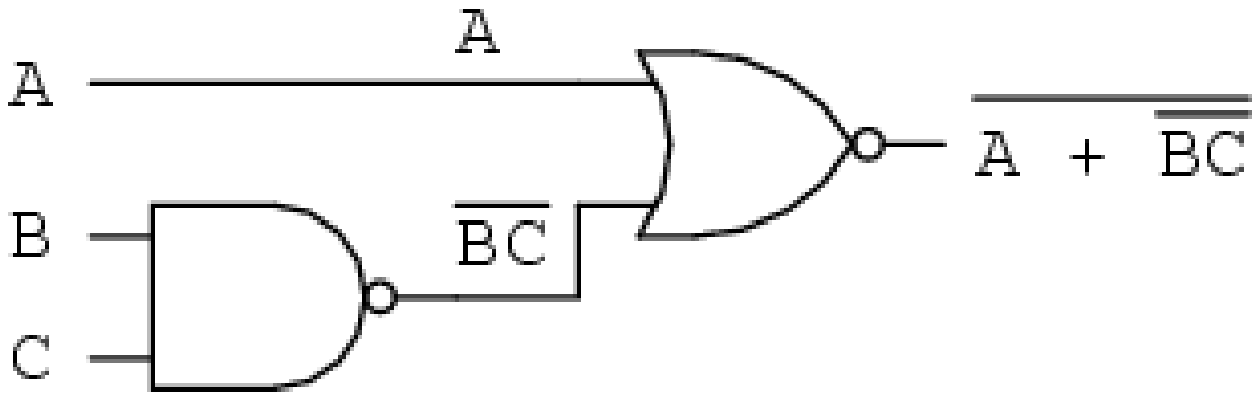


NAND to *Negative-OR*

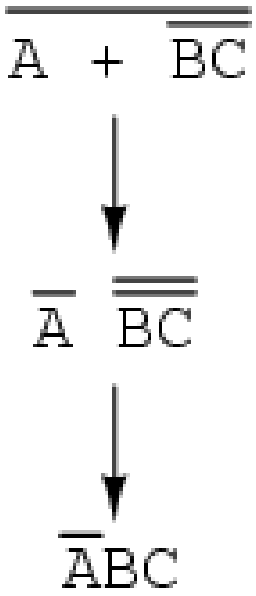


NOR to *Negative-AND*

Example: consider the expression (A + (BC)')' and reduce it using DeMorgan's Laws:



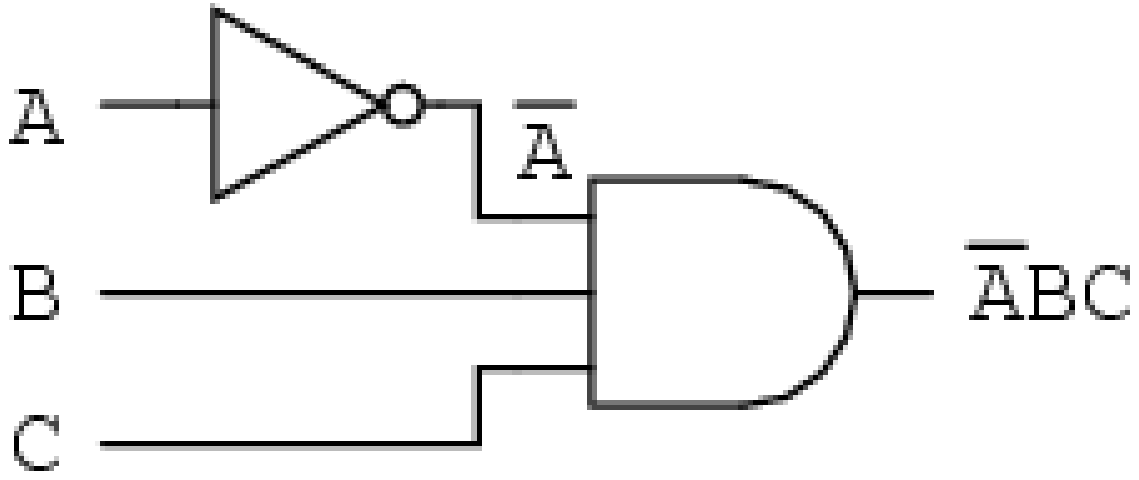
Following the advice of breaking the longest (uppermost) bar first



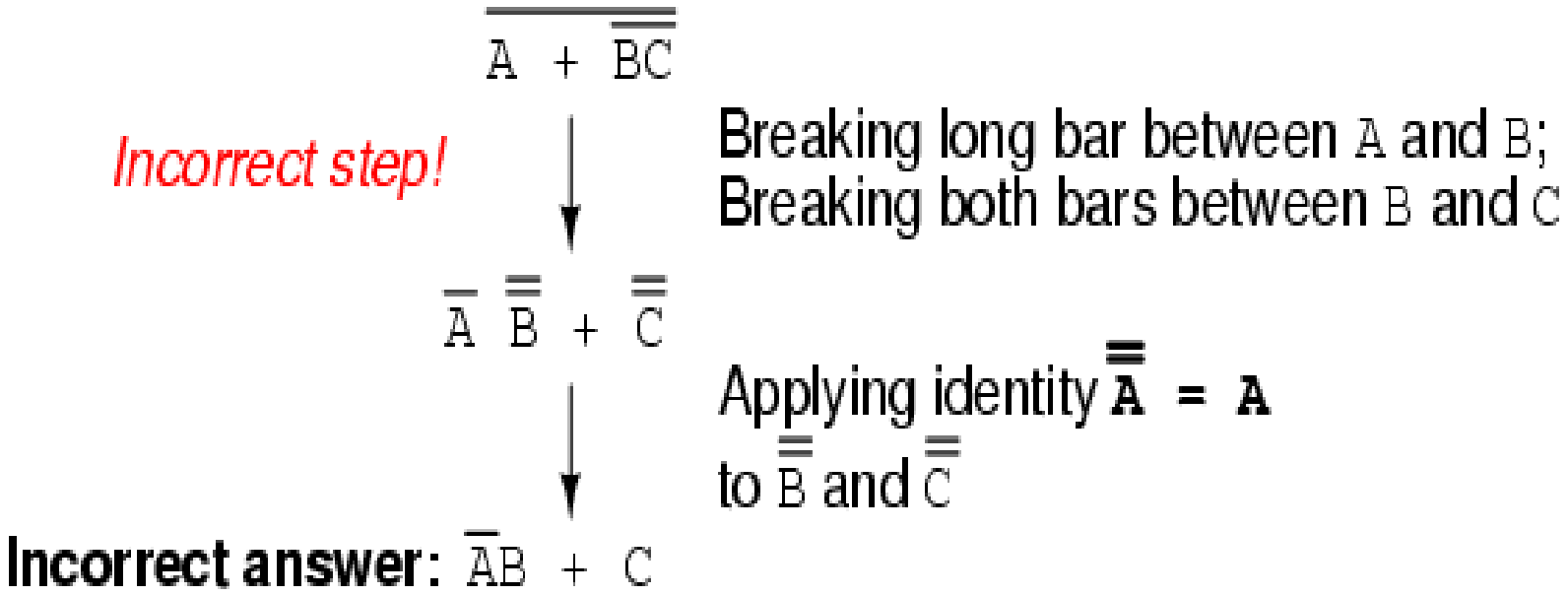
Breaking longest bar
(addition changes to multiplication)

Applying identity $\overline{\overline{A}} = A$
to $\overline{\overline{BC}}$

As a result, the original circuit is reduced to a three-input AND gate with the A input inverted:

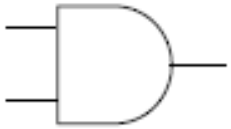


You should *never* break more than one bar in a single step, as illustrated here:

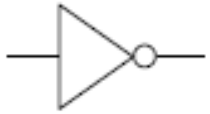


COMP2 - Jan 2013

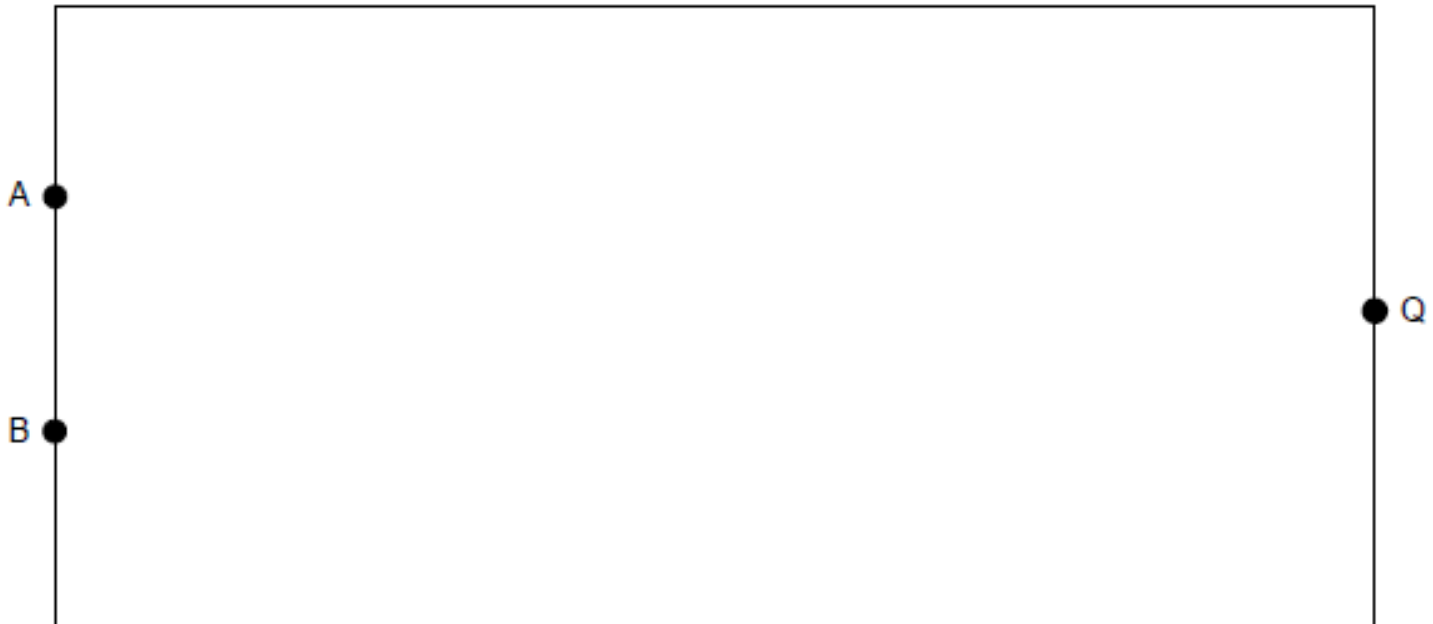
4 (a) Represent the Boolean equation $Q = \overline{\overline{A}.\overline{B}}$ as a logic circuit by drawing a diagram in the space below using **only** the following symbols:

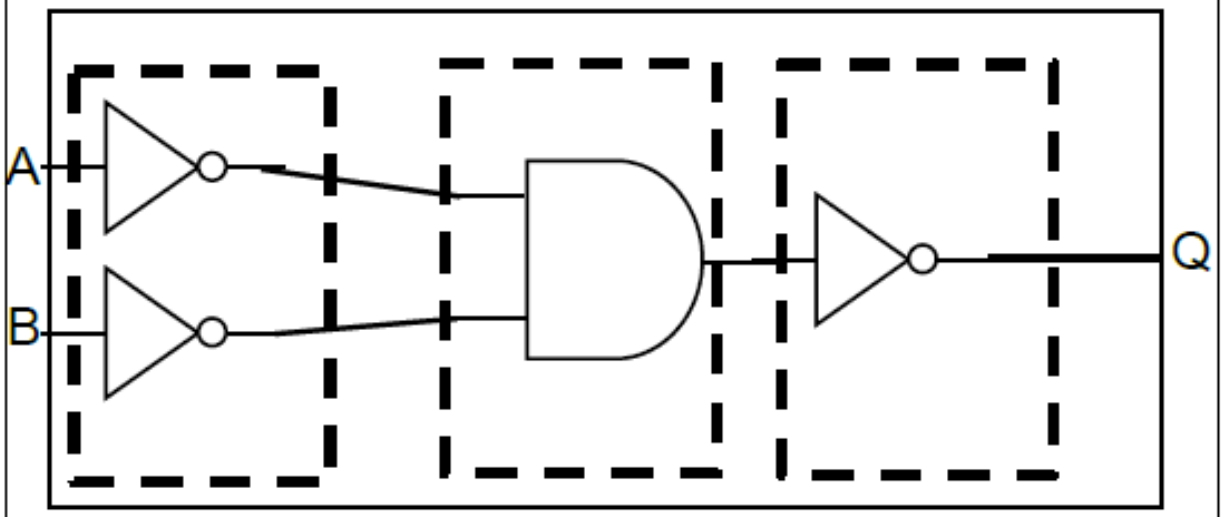


AND



NOT



4	**a**		 1 mark – logic of first part satisfies NOT A, NOT B; 1 mark – inputs into an AND gate; 1 mark – output from AND gate passes through a NOT gate and connected to Q;

4 (b) Use the following truth tables to demonstrate that $A + B = \overline{\overline{A}.\overline{B}}$

A	B	A + B
0	0	
0	1	
1	0	
1	1	

A	B	$\overline{A}$	$\overline{B}$	$\overline{A}.\overline{B}$	$\overline{\overline{A}.\overline{B}}$
0	0				
0	1				
1	0				
1	1				

(4 marks)

4 (c) What is the name commonly associated with the statement $A + B = \overline{\overline{A}.\overline{B}}$?

.....

(1 mark)

4	b	

A	B	A + B
0	0	0
0	1	1
1	0	1
1	1	1

1 mark for correct A + B column;

A	B	$\overline{A}$	$\overline{B}$	$\overline{A} . \overline{B}$	$\overline{\overline{A} . \overline{B}}$
0	0	1	1	1	0
0	1	1	0	0	1
1	0	0	1	0	1
1	1	0	0	0	1

1 mark for columns $\overline{A}$ and $\overline{B}$ being correct;
1 mark for $\overline{A} . \overline{B}$ column being correct;
1 mark for $\overline{\overline{A} . \overline{B}}$ column being correct;