IGCSE Computer Science

30 /31

Name: _____

Chapter 10 Test: Boolean Logic

Q1

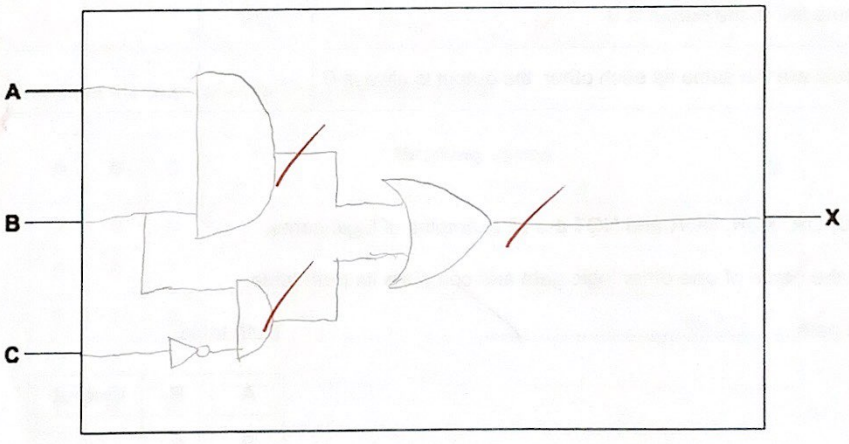
Consider this logic expression.

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

(a) Draw a logic circuit for this logic expression.

Each logic gate must have a maximum of **two** inputs.

Do **not** simplify this logic expression.



4 [4]

(b) Complete the truth table from the given logic expression.

A	B	C	Working space	X
0	0	0	0 0	0
0	0	1	0 0	0
0	1	0	0 1	1
0	1	1	0 0	0
1	0	0	0 0	0
1	0	1	0 0	0
1	1	0	1 1	1
1	1	1	1 0	1

4 [4]

Q2

7 NAND, OR and XOR are three types of logic gate.

(a) **Four** statements are shown about the logic gates.

Tick (✓) to show which statements apply to each logic gate. Some statements may apply to more than one logic gate.

Statement	NAND (✓)	OR (✓)	XOR (✓)
if both inputs are 1, the output is 1		✓	
if both inputs are different from each other, the output is 1	✓	✓	✓
if both inputs are 0, the output is 0	XOR	✓	✓
if both inputs are the same as each other, the output is always 0			✓

[4]

(b) NAND, OR, XOR, NOR and NOT are all examples of logic gates.

State the name of **one** other logic gate and complete its truth table.

Logic gate ..... AND .....

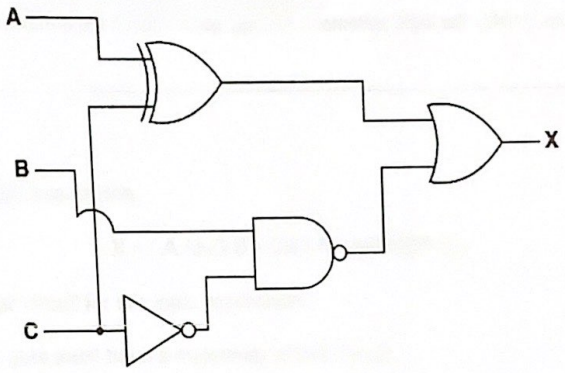
Truth table:

A	B	Output
0	0	0
0	1	0
1	0	0
1	1	1

[2]

Q3)

Consider the logic circuit:



(a) Write a logic statement to match the given logic circuit.

(A XOR C) OR (B NAND (NOT C)) 3/ [3]

(b) Complete the truth table for the given logic circuit.

A	B	C	Working space (A XOR C ; B NAND NOT C)	X
0	0	0	0 ; 1	1
0	0	1	1 ; 1	1 ✓
0	1	0	0 ; 0	0 ✓
0	1	1	1 ; 1	1
1	0	0	1 ; 1	1 ✓
1	0	1	0 ; 1	1 ✓
1	1	0	1 ; 0	1
1	1	1	0 ; 1	1

4 [4]

Q4)

8 Consider the following logic statement:

X = (((A AND NOT B) OR (NOT (B NOR C))) AND C)

(a) Draw a logic circuit to represent the given logic statement.

Do **not** attempt to simplify the logic statement. All logic gates must have a maximum of **two** inputs.

[6]

(b) Complete the truth table for the given logic statement.

A	B	C	Working space			X
			AND B'	NOT(B NOR C)	OR	
0	0	0	0	0	0	0
0	0	1	0	1	1	1
0	1	0	0	1	1	0
0	1	1	0	1	1	1
1	0	0	1	0	1	0
1	0	1	1	1	1	1
1	1	0	0	1	1	0
1	1	1	0	1	1	1

[4]