Past-paper walk-through

Boolean Algebra and Logic Circuits ▪ 15.2

eXercise

Questions in this set

- 9618/32 N25 Q6 ▪ 9 marks
- 9618/33 N25 Q6 ▪ 8 marks
- 9618/32 N24 Q6 ▪ 9 marks
- 9618/33 N24 Q7 ▪ 10 marks

Reading the mark scheme

- **B1** a mark that stands on its own – the point is either made or not.
- **M1** a *method* mark: the right approach, even if the number is wrong.
- **A1** an *answer* mark, and it usually needs its M mark first.
- **C1** a working mark on the way to the answer.
- // separates answers the examiner accepts equally.
- **ecf** = error carried forward: a later part is marked on your own earlier answer.
- **owtte** = "or words to that effect" – the wording need not match.

Question 6

9618/32 N25 ▪ Q6 ▪ 9 marks

The diagram shows a logic circuit.

(a) Complete the truth table for the given logic circuit. Show your working.

			Working space				
A	**B**	**C**	**P**	**Q**	**R**	**S**	**Z**
0	0	0					
0	0	1					
0	1	0					
0	1	1					
1	0	0					
1	0	1					
1	1	0					
1	1	1					

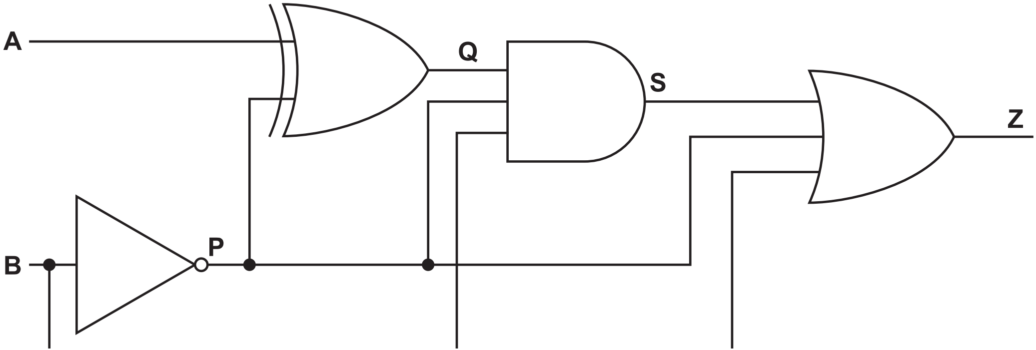
Question 6

[3]

Question 6

			Working space				
A	**B**	**C**	**P**	**Q**	**R**	**S**	**Z**
0	0	0					
0	0	1					
0	1	0					
0	1	1					
1	0	0					
1	0	1					
1	1	0					
1	1	1					

Question 6



Question 6

A \ BC	00	01	11	10
0				
1				

Question 6

9618/32 N25 ▪ Q6 ▪ 9 marks

The diagram shows a logic circuit.

(b)(i) Complete the Karnaugh map (K-map) for the Boolean expression:

$$\mathbf{A.B.C + A.B.C + A.B.C + A.B.C}$$

A \ BC	**00**	**01**	**11**	**10**
0				
1				

Question 6

[2]

(b)(ii) Draw loop(s) around appropriate group(s) in the K-map to produce an optimal sum-of-products. **[2]**

(b)(iii) Write the Boolean expression from your answer to part **b(ii)** as a simplified sum-of-products. Do **not** carry out any further simplification. **[2]**

Solution 6

(a) Worked steps

The working columns are worth a mark on their own, so fill them in rather than jumping to Z.

- Give every intermediate signal its own column: P, Q, R, S, then Z.
- Work **gate by gate, left to right**: each column depends only on columns already filled.
- Do all eight rows of one column before starting the next. Going row by row across the whole circuit is where mistakes creep in, and one wrong intermediate value corrupts every Z beneath it.
- Read each gate carefully: an inverting output (NAND, NOR) is the opposite of the gate it looks like, and a small circle on the symbol is the only thing telling you so.

Solution 6

Mark scheme

- - One mark for working, (all four columns P, Q, R and S)
- - One mark for first four rows of column Z
- - One mark for second four rows of column Z
- Working space
- A
- B
- C
- P

Solution 6

- Q
- R
- S
- Z
- 0
- 0
- 0
- 1
- 1
- 0

Solution 6

- 0
- 1
- 0
- 0
- 1
- 1
- 1
- 1
- 1
- 1

Solution 6

- 0
- 1
- 0
- 0
- 0
- 1
- 0
- 1
- 0
- 1

Solution 6

- 1
- 0
- 0
- 0
- 0
- 0
- 1
- 0
- 0
- 1

Solution 6

- 0
- 0
- 0
- 1
- 1
- 0
- 1
- 1
- 0
- 1

Solution 6

- 0
- 1
- 1
- 1
- 0
- 0
- 1
- 1
- 0
- 1

Solution 6

- 1
- 1
- 1
- 0
- 1
- 0
- 0
- 0

Solution 6

(b)(i) Mark scheme

- Two marks if no errors present
- One mark if only one error present
- BC
- A
- 00
- 01
- 11
- 10

Solution 6

- 0
- 0
- 1
- 1
- 0
- 1
- 0
- 1
- 0
- 1

Solution 6

- 2
- October/November
- 2025

Solution 6

(b)(ii) Worked steps

- Loop only 1s, in groups of 1, 2, 4 or 8 — never 3 or 6.
- Make each loop as large as it can be, and let loops overlap where that allows a bigger one.
- Remember the map wraps left-to-right and top-to-bottom.
- Use the fewest loops that cover every 1; each loop is one term of the sum-of-products, made of the variables that stay constant across it.

Mark scheme

- One mark for each correct loop (Max 2)

Solution 6

- BC
- A
- 00
- 01
- 11
- 10
- 0
- 0
- 1
- 1

Solution 6

- 0
- 1
- 0
- 1
- 0
- 1

Solution 6

(b)(iii) Mark scheme

- One mark for each mark point (Max 2)
- ▪ One correct Boolean term with a + / OR sign
- ▪ All Boolean terms and operators correct and no other terms present
- _ _ _
- A.C + B.C + A.B.C
- 2
- October/November
- 2025

Question 6

9618/33 N25 ▪ Q6 ▪ 8 marks

The Karnaugh map (K-map) represents a logic circuit with four inputs.

(a)(i) Draw loop(s) around appropriate group(s) in the K-map to produce an optimal sum-of-products. **[3]**

(a)(ii) Write the Boolean expression from your answer to part **a(i)** as a simplified sum-of-products. Do **not** carry out any further simplification. **[2]**

(b) Simplify the following expression using De Morgan's laws and Boolean algebra. Show all the stages in your simplification.

$$X = \overline{A + B + C} + \overline{\overline{B} + C}$$

[3]

Question 6

CD \ AB	00	01	11	10
00	0	0	0	0
01	1	1	1	0
11	0	1	1	1
10	0	0	1	1

Solution 6

(a)(i) Worked steps

Looping a Karnaugh map is mechanical once the rules are followed in order.

- Loop only **1s**, in groups whose size is a power of two: 8, then 4, then 2, then 1.
- Make every loop as **large** as possible, even where loops overlap — overlapping is allowed and is usually what makes the expression optimal.
- The map **wraps**: the left and right columns are adjacent, and so are the top and bottom rows, so a group can straddle an edge.
- Stop when every 1 is inside at least one loop, using as few loops as possible.

Each loop becomes one product term, made of the variables that do **not** change across it. **Mark scheme**

Solution 6

- One mark for each correct loop (Max 3)
- CD
- 00
- 01
- 11
- 10
- 00
- 0
- 0
- 0

Solution 6

- 0
- 01
- 1
- 1
- 1
- 0
- 11
- 0
- 1
- 1

Solution 6

- 1
- 10
- 0
- 0
- 1
- 1

Solution 6

(a)(ii) Mark scheme

- One mark per mark point (Max 2)
- MP1
- One correct Boolean term with an OR sign
- MP2
- All three correct Boolean terms connected by OR signs and no other terms present.
- A.C.D + B.D + A.C
- 2

Solution 6

- AB

Solution 6

(b) Worked steps

Show every stage — the marks are for the laws used, not for the final line.

- Apply **De Morgan's** to each bar over a sum: $\overline{A+B+C} = \bar{A} \cdot \bar{B} \cdot \bar{C}$, and $\overline{B+C} = \bar{B} \cdot \bar{C}$.
- That gives $X = \bar{A}\bar{B}\bar{C} + \bar{B}\bar{C}$.
- Now factorise: $\bar{B}\bar{C}$ is common, so $X = \bar{B}\bar{C}(\bar{A}+1)$.
- Since $\bar{A}+1=1$, this simplifies to $X = \bar{B}\bar{C}$.

The pattern worth learning: whenever one term is contained in another, the longer one is absorbed — that is the absorption law, and it is what the second mark is looking for.

Mark scheme

Solution 6

- One mark for correct use of De Morgan's laws
- One mark for correct use of any other Boolean algebra law
- X = A+B+C + B+C
- (X=) A.B.C + B.C …………………………….(De Morgan's)
- (X=) A.B.C + B.C …………………………….(Double negation)
- (X=) C.(A.B + B) ……………………………..(Distributive)
- One mark for correct answer
- (X=) C.(A + B) ……………………………..(Redundancy)

Question 6

9618/32 N24 ▪ Q6 ▪ 9 marks

6 The truth table for a logic circuit is shown.

INPUT				OUTPUT
A	**B**	**C**	**D**	**X**
0	0	0	0	0
0	0	0	1	1
0	0	1	0	1
0	0	1	1	0
0	1	0	0	0
0	1	0	1	1
0	1	1	0	1

Question 6

0	1	1	1	0
1	0	0	0	0
1	0	0	1	1
1	0	1	0	1
1	0	1	1	0
1	1	0	0	0
1	1	0	1	1
1	1	1	0	1
1	1	1	1	0

(a) Write the Boolean logic expression that corresponds to the given truth table as the sum-of-products.

Question 6

sum of products.

Question 6

Question 6

(b) Complete the Karnaugh map (K-map) for the given truth table.

CD \ AB	00	01	11	10
00				
01				
11				
10				

[2]

Question 6

Question 6

(c) Draw loop(s) around appropriate group(s) in the K-map to produce an optimal sum-of-products. [2]

(d) Write the Boolean logic expression from your answer to part **(c)** as the simplified sum-of-products.

Solution 6

(a) Worked steps

A sum-of-products expression is read straight off the truth table: **one product term for every row whose output is 1**, and nothing else.
For each such row, write the four inputs multiplied together, using the plain letter where the input is 1 and the complemented letter where it is 0. Then add the terms with +.

- Eight rows give 1, so there are exactly **eight** product terms.
- Every term contains all four variables — `A`, `B`, `C` and `D`, each either plain or barred.
- Join them with +, the Boolean OR.

Solution 6

The mark scheme credits partly: all eight and no extras scores 3, five to seven scores 2, three or four scores 1. So work down the table in order and tick each 1-row off as you write it — the usual loss is a missed row, not a wrong term.

Solution 6

Do **not** simplify here. That is what parts (b) to (d) are for, and an already-simplified answer cannot show the eight products this part is marked on.

Mark scheme

- Three marks for all eight correct products and no additional products
- Two marks for five, six or seven correct products
- One mark for three or four correct products
- $(X =)\ A.\ B.\ C.\ D + A.\ B.\ C.\ D + A.\ B.\ C.\ D. + A.\ B.\ C.\ D + A.\ B.\ C.\ D + A.\ B.\ C.\ D + A.\ B.\ C.\ D + A.\ B.\ C.\ D$
- 3

Solution 6

(b) Worked steps

The Karnaugh map holds the same information as the truth table, laid out so that neighbouring cells differ in one variable only.

- The columns are labelled `AB` and the rows `CD` (or the other way round — follow the printed map), and both run in **Gray-code order 00, 01, 11, 10**, not in binary counting order.
- Put a 1 in the cell for every product term from (a), and a 0 everywhere else.

Two marks with no errors, one with a single error. The Gray-code ordering is what earns them: writing the headings 00, 01, 10, 11 puts the 1s in the wrong cells and breaks every grouping in part (c), even though the map looks plausible. **Mark scheme**

Solution 6

- Two marks if no errors present
- One mark if one error present
- AB
- CD
- 00
- 01
- 11
- 10
- 00
- 0

Solution 6

- 0
- 0
- 0
- 01
- 1
- 1
- 1
- 1
- 11
- 0

Solution 6

- 0
- 0
- 0
- 10
- 1
- 1
- 1
- 1
- 2

Solution 6

(c) Worked steps

Loop the 1s in the largest possible blocks. The rules are exact:

- Groups must be **rectangular** and contain a number of cells that is a **power of two** — 8, 4, 2, then 1 as a last resort.
- Bigger is better: each doubling of a group removes one variable from its term.
- Groups **may overlap**, and they **wrap around** the edges of the map — the left column is adjacent to the right, and the top row to the bottom.
- Every 1 must be inside at least one group; no group may contain a 0.

Solution 6

One mark per correct loop, to a maximum of two. Overlapping is not a mistake and wrapping is not cheating; both are how the map's adjacency works, and a candidate who avoids them ends up with more, smaller groups and a longer expression. **Mark scheme**

- One mark for each correct loop (Max 2)
- AB
- CD
- 00
- 01
- 11

Solution 6

- 10
- 00
- 0
- 0
- 0
- 0
- 01
- 1
- 1
- 1

Solution 6

- 1
- 11
- 0
- 0
- 0
- 0
- 10
- 1
- 1
- 1

Solution 6

- 1
- 2
- 9618/32
- October/November 2024

Solution 6

(d) Worked steps

Turn each loop into a term, then add the terms.

- For a loop, keep only the variables that stay the **same** across every cell in it; the ones that change are eliminated.
- Write each kept variable plain if it is 1 throughout the loop, barred if it is 0.
- Join the loops with +.

Here that gives $X = \overline{C}.D + C.\overline{D}$ — two terms, each of two variables, because each loop spans four cells and so drops two of the four variables.

Solution 6

The two marks are for a **correct term** and for having **only** the correct terms joined by the right operator. A leftover variable means the loop was drawn too small; an extra term means a group was drawn that some other group already covered.

Solution 6

Incidentally, $\overline{C}.D + C.\overline{D}$ is exactly $C \oplus D$ — the exclusive-OR of the two inputs, which is worth recognising when it appears.

Mark scheme

- One mark for each mark point (Max 2)
- MP1 Any correct relevant Boolean term
- MP2 Boolean terms with correct operator + and no other terms present
- (X =) C. D + C. D // C. D. + C. D
- 2

Question 7

9618/33 N24 ▪ Q7 ▪ 10 marks

7 The truth table for a logic circuit is shown.

INPUT				OUTPUT
A	**B**	**C**	**D**	**T**
0	0	0	0	0
0	0	0	1	1
0	0	1	0	0
0	0	1	1	1
0	1	0	0	0
0	1	0	1	0
0	1	1	0	0

Question 7

0	1	1	1	0
1	0	0	0	0
1	0	0	1	1
1	0	1	0	0
1	0	1	1	1
1	1	0	0	0
1	1	0	1	1
1	1	1	0	0
1	1	1	1	1

(a) Write the Boolean logic expression that corresponds to the given truth table as the sum-of-products.

Question 7

Question 7

Question 7

(b) Complete the Karnaugh map (K-map) for the given truth table.

CD \ AB	00	01	11	10
00				
01				
11				
10				

[2]

Question 7

Question 7

(c) Draw loop(s) around appropriate group(s) in the K-map to produce an optimal sum-of-products. [2]

(d) **(i)** Write the Boolean logic expression from your answer to part **(c)** as the simplified sum-of-products.

Question 7

(ii) Use Boolean algebra to write your answer to part **(d)(i)** in its simplest form.

Solution 7

(a) Worked steps

A sum-of-products expression is read straight off the truth table: **one product term for every row whose output is 1**, and nothing else.

- For each such row, write all four variables multiplied together — the plain letter where the input is 1, the complemented letter where it is 0.
- Join the terms with +.

Six rows give 1 here, so there are six product terms and each contains all four variables.

Solution 7

Three marks, one for **every two correct products**, so the marking is generous to a partly complete answer but unforgiving of extra terms. Work down the table in order and tick each 1-row off as you write it.

Do not simplify at this stage. Parts (b) to (d) are the simplification, and a pre-simplified answer cannot show the six products this part is marked on.

Mark scheme

- One mark for every two correct products (Max 3)
- $(T =)$ A. B. C. D + A. B. C. D + A. B. C. D + A. B. C. D + A. B. C. D + A. B. C. D
- 3
- 9618/33
- October/November 2024

Solution 7

(b) Worked steps

Transfer the same information to a Karnaugh map. Two marks with no errors, one with a single error.

- The rows and columns are labelled **AB** and **CD**, and both run in **Gray-code order: 00, 01, 11, 10** — not in binary counting order.
- Put a 1 in the cell for each product term from (a), and 0 everywhere else.

The Gray-code ordering is what earns these marks and what makes part (c) possible: it is the reason neighbouring cells differ in exactly one variable. A map headed 00, 01, 10, 11 puts the 1s in the wrong cells, and every grouping drawn on it is then wrong too — while the map still looks entirely plausible. **Mark scheme**

Solution 7

- Two marks if no errors present
- One mark if one error present
- AB
- CD
- 00
- 01
- 11
- 10
- 00
- 0

Solution 7

- 0
- 0
- 0
- 01
- 1
- 0
- 1
- 1
- 11
- 1

Solution 7

- 0
- 1
- 1
- 10
- 0
- 0
- 0
- 0
- 2

Solution 7

(c) Worked steps

Loop the 1s in the largest rectangular blocks possible, with cell counts that are powers of two — 8, then 4, then 2.

- Groups **may overlap**, and they **wrap around** the edges of the map.
- Every 1 must be in at least one group; no group may contain a 0.
- Bigger groups are better, because each doubling removes one variable from the resulting term.

One mark per correct loop, to a maximum of two. **Mark scheme**

- One mark for each correct loop (Max 2)

Solution 7

- AB
- CD
- 00
- 01
- 11
- 10
- 00
- 0
- 0
- 0

Solution 7

- 0
- 01
- 1
- 0
- 1
- 1
- 11
- 1
- 0
- 1

Solution 7

- 1
- 10
- 0
- 0
- 0
- 0
- 2
- 9618/33
- October/November 2024

Solution 7

(d)(i)

Worked steps

Turn each loop into a term: keep only the variables that are **constant across the whole loop**, plain where they are 1 and barred where they are 0, and add the terms together.

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

Two marks: one for any correct Boolean term, one for having the correct terms with the right operator and **no others**.
Each loop here spans four cells, so two of the four variables vary within it and drop out — which is why each term has two variables rather than four.

Mark scheme

- One mark for each mark point (Max 2)
- ▪

Solution 7

(d)(ii) Worked steps

One further step, and it is a **factorisation** rather than another map.

$$T = A \cdot \overline{D} + B \cdot \overline{D} = \overline{D} \cdot (A + B)$$

One mark, for the simplest form.

Solution 7

$\overline{D}$ is common to both terms, so the distributive law takes it outside the bracket. The result uses one fewer gate than the sum-of-products version, which is the practical point of the whole question: a Karnaugh map gets you to a minimal sum of products, and ordinary algebra can sometimes go one step further.

Mark scheme

- One mark for simplest form (Max 1)
- (T =) D. (A. B)
- 1