

AQA Computer Science A-Level
4.6.5 Boolean algebra
Concise Notes



Specification:

4.6.5.1 Using Boolean algebra:

Be familiar with the use of Boolean identities and De Morgan's laws to manipulate and simplify Boolean expressions.



Boolean algebra

- Concerns **representing values with letters** and **simplifying expressions**.
- Uses the Boolean values **TRUE** and **FALSE**

Notation

Expression	Meaning
$A, B, C, \text{ etc.}$	An unknown Boolean value
\bar{A}	NOT A.
$A \cdot B$	A AND B
AB	A AND B
$A + B$	A OR B

Order of precedence

- Some operations must be **applied before others**

Operator	Precedence
Brackets	Highest
NOT	·
AND	·
OR	·
	Lowest

Boolean identities

$$A \cdot 0 = 0$$

$$B \cdot 1 = B$$

$$C \cdot C = C$$

$$E + 1 = 1$$

$$F + F = F$$

$$D + 0 = D$$

$$\overline{\overline{G}} = G$$

$$1 + 1 = 1$$



De Morgan's laws

“Break the bar and change the sign”

- “The bar” refers to an [underline](#) representing the NOT operation
- “The sign” refers to changing between + (OR) and • (AND)
- Can also be applied in reverse, by [changing the sign](#) and [building the bar](#)

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

$$\bar{C} + \bar{D} = \overline{C \cdot D}$$

Distributive rules

- Similar to [expanding brackets](#) in Mathematics

$$A \cdot (B + C) = A \cdot B + A \cdot C$$

$$(A + B) \cdot (C + D) = AC + AD + BC + BD$$

