

Exercise walk-through

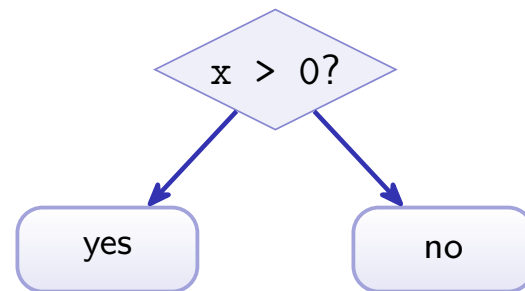
2.2

Boolean Expressions

AP Computer Science A

What we will cover

- 1 Boolean expressions
- 2 The if statement
- 3 Compound & nested conditions
- 4 while & for loops
- 5 Nested iteration & run-time



The integer patterns the exam tests directly

reading digits

repeatedly take $n \% 10$ for the **last digit**, then $n /= 10$ to drop it
– loop while $n > 0$

why it works

$\%$ gives the remainder and integer $/$ truncates, so together they peel one digit per pass

a running total

declare the accumulator **outside** the loop, add inside it

a counter

the same shape, but incrementing on a condition rather than adding a value

a flag

a boolean recording **whether something happened** – set inside the loop, tested after it

Total, count, max/min and flag cover most free-response loop questions. Decide which the wording is asking for before writing anything.

Exam tips

- Get boundary conditions right: use `<` vs `<=` deliberately, and watch the first and last iteration of every loop (off-by-one is the classic bug).
- Build compound conditions with `&&`, `||`, `!` and remember **short-circuit** evaluation (put the null check first).
- Trace nested loops by counting how many times the **inner** body runs in total.
- Choose the right structure – `if/else if` for ranges, a loop for repetition – and avoid an infinite loop by updating the loop variable.
- Apply **De Morgan's laws** when you simplify or negate a boolean condition.

Exercises 2.1-2.3

2.1 State what a Boolean expression evaluates to.

[1]

2.2 State the difference between `==` and `=`.

[2]

2.3 Evaluate `7 >= 7`.

[1]

Exercises 2.1-2.3

2.1 State what a Boolean expression evaluates to.

[1]

2.1

- either `true` or `false`
-

2.2 State the difference between `==` and `=`.

[2]

2.2

- `==` tests whether two values are equal; `=` assigns a value to a variable
-

2.3 Evaluate `7 >= 7`.

[1]

2.3

- `true`

Exercises 3.1-3.2

3.1 Which operator tests equality?

[1]

☐

A =

B ==

C !=

D <=

3.2 4 != 4 evaluates to

[1]

☐

A true

B false

C 4

D an error

Exercises 3.1-3.2

3.1 Which operator tests equality?

[1]

☐

A =

B ==

C !=

D <=

3.1 B

- ==.

3.2 4 != 4 evaluates to

[1]

☐

A true

B false

C 4

D an error

3.2 B

- false.

Exercise 3.3 (a)

3.3 Given `int x = 6, y = 9;`:

(a) evaluate `x < y`.

[1]

Exercise 3.3 (a)

3.3 Given `int x = 6, y = 9;`:

(a) evaluate `x < y`.

[1]

(a)

- `true`

Exercise 3.3 (b)

3.3 Given `int x = 6, y = 9;`:

(b) evaluate `x == y`.

[1]

Exercise 3.3 (b)

3.3 Given `int x = 6, y = 9;`:

(b) evaluate `x == y`.

[1]

(b)

- `false`

Exercise 3.3 (c)

3.3 Given `int x = 6, y = 9;`:

(c) evaluate `x != y`.

[1]

Exercise 3.3 (c)

3.3 Given `int x = 6, y = 9;`:

(c) evaluate `x != y`.

[1]

(c)

- `true`

Exercise 3.4

3.4 A Java programme defines `int score = 60;` and `boolean pass = score >= 60;`. State `pass`, compare with the result of `score > 60`, and give two neighbouring integer inputs useful for checking this boundary. [4]

Exercise 3.4

3.4 A Java programme defines `int score = 60;` and `boolean pass = score >= 60;`. State `pass`, compare with the result of `score > 60`, and give two neighbouring integer inputs useful for checking this boundary. [4]

3.4

`pass` is `true`; `score > 60` is `false`; inputs 59 and 61; these lie immediately below and above the boundary, checking exclusion and inclusion respectively.