

Exercise walk-through

Informal Run-Time Analysis ■ 2.12

eXercise

You must be able to

- count how many times a statement executes as a measure of a program's work
- understand that a loop over n items runs its body about n times
- explain why a **nested loop** over n items runs its inner body about n^2 times

Method — Counting executions

We estimate a program's work by counting how many times a statement runs.

- A single loop over n items runs its body about n times.
- A **nested loop** over n items runs its inner body about $n \times n = n^2$ times.

So nested loops grow much faster than single loops as n increases.

Method — Worked count

For a loop that runs n times **inside** a loop that runs n times, the inner statement runs n^2 times. If $n = 5$ that is 25 executions; if n doubles to 10 it jumps to 100 – four times the work. A single loop would only double.

Practice 2.1 ■ 1 mark

State roughly how many times a single loop over n items runs its body.

Solution 2.1

Method: Counting executions

Worked steps

about n times **B1**.

Practice 2.2 ■ 1 mark

State roughly how many times a nested loop over n items runs its inner body.

Solution 2.2

Method: Counting executions

Worked steps

about n^2 times **B1**.

Practice 2.3 ■ 1 mark

A single loop runs over 100 items. State roughly how many times its body runs.

Solution 2.3

Method: Counting executions

Worked steps

about 100 times **B1**.

Exam-style 3.1 ■ 1 mark

A single loop over n items runs its body about times.

A 1

B n

C n^2

D 2^n

Solution 3.1

Method: Counting executions

A 1

B n



C n^2

D 2^n

Worked steps

B.

Exam-style 3.2 ■ 1 mark

A nested loop over n items runs its inner body about times.

A n

B n^2

C $\log n$

D 1

Solution 3.2

Method: Counting executions

A n

B n^2



C $\log n$

D 1

Worked steps

B.

Exam-style 3.3 ■ 1 mark

A nested loop processes $n = 10$ items.

- (a) State roughly how many inner-body runs there are.
- (b) State whether this grows faster than a single loop.
- (c) State roughly how many runs a single loop over 10 items has.

Solution 3.3

Method: Counting executions

Worked steps

(a) about 100 (10^2) **B1**. (b) yes **B1**. (c) about 10 **B1**.