



F2.7 Loop patterns

Decisions and loops · GCSE · OCR J277 2.2.1, AQA 8525 3.2.2, Edexcel
1CP2 1.2.4 · about 20 min

What this lesson is about

Running totals, counting, finding the largest, nested loops and trace tables.

- Running total
- Counting
- Finding the largest
- Loops inside loops
- Tracing a loop

Questions

7 marks in all

1. What does this program print?

[1 mark]

```
total = 0
for d in [40, 35, 30]:
    total = total + d
print(total, total / 3)
```

Answer:

105 35.0

A running total adds each value: 105. Dividing by how many gives the average, 35.0.

2. What does this program print?

[1 mark]

```
count = 0
for d in [50, 20, 35, 10]:
    if d < 30:
        count = count + 1
print(count)
```

Answer:

2

Only 20 and 10 are under 30, so the count is 2.

3. What does this program print?

[1 mark]

```
most = 0
for d in [12, 51, 30]:
    if d > most:
        most = d
print(most)
```

Answer:

51

most is replaced whenever a bigger value turns up, so it ends as the largest, 51.

4. An outer loop runs 3 times and the loop inside it runs 4 times each time. How many times does the inner block run?

[1 mark]

Answer: 12. The inner loop runs all the way through for every time round the outer loop: 3 times 4.

5. A program puts `total = 0` inside the loop instead of before it. What goes wrong?

[1 mark]

- ☐ A The total is reset every time round, so it ends as just the last value
- ☐ B Nothing
- ☐ C A syntax error
- ☐ D The loop never ends

Answer: A. The starting value belongs before the loop, so it is set once.

6. What is a trace table for?

[1 mark]

- ☐ A Following the values of variables step by step as an algorithm runs
- ☐ B Drawing the robot's path
- ☐ C Listing a program's errors
- ☐ D Storing data in a program

Answer: A. One column per variable, one row each time something changes: it shows exactly what an algorithm does.

7. What does this program print?

[1 mark]

```
for row in range(1, 3):
    for col in range(1, 4):
        print(row, col)
```

Answer:

1 1
1 2
1 3
2 1
2 2
2 3

For each row, the inner loop goes through all three columns before the row changes.

The task: times grid

Print a multiplication grid of five rows and five columns with two nested `for` loops. Each row is the numbers separated by single spaces, so the first row is `1 2 3 4 5` and the last is `5 10 15 20 25`. Work the numbers out; do not type them. The robot must not drive.

```
# the two lines every program starts with: the commands, then the robot
from bugbot import *
connect()

for row in range(1, 6):
    print(row)
```

The hint students can ask for: One loop for the rows, and another inside it for the columns. Print without starting a new line inside the inner loop, then start one when each row finishes.

A solution

```
from bugbot import *
connect()
for row in range(1, 6):
    for col in range(1, 6):
        print(row * col, end=' ')
    print()
```

Any program that meets the task's checks is marked correct in the simulator; this is one way, not the only way.