



F13.3 Pseudocode in the exam

What this lesson is about

Reading and writing exam pseudocode and flowcharts, and translating both ways.

- The same algorithm, three ways
- Reading it
- Writing it
- Flowcharts

Questions

5 marks in all

1. How many times does FOR i = 1 TO 5 run?

[1 mark]

- ☐ A 5
- ☐ B 4
- ☐ C 6
- ☐ D It depends on the board

Answer: A. The end value is included, unlike range(1, 5) in Python.

2. Which pseudocode keyword prints something?

[1 mark]

Answer: OUTPUT. OUTPUT is print.

3. In a flowchart, what does a diamond mean?

[1 mark]

- ☐ A A decision
- ☐ B A process
- ☐ C Input or output
- ☐ D The start

Answer: A. Each way out of a diamond is labelled.

4. What does this program print?

[1 mark]

```
for i in range(1, 4):  
    print(i * i)
```

Answer:

1
4
9

FOR 1 TO 3 includes 3, so range stops at 4.

5. You are asked to write an algorithm. What may you use?

[1 mark]

- ☐ A Pseudocode or a real language, as long as it is clear
- ☐ B Only your board's reference language
- ☐ C Only Python
- ☐ D Only a flowchart

Answer: A. The logic earns the marks, not the syntax.

The task: pseudocode into Python

Turn the pseudocode in the comment into a working program. It asks for a number of steps, drives that many 10 cm steps while counting the total distance, and prints the total. `input()` is answered for you with `4`.

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

# steps = USERINPUT
# total = 0
# FOR i = 1 TO steps
#     MOVE FORWARD 10
#     total = total + 10
#     OUTPUT "step " + i + ": " + total
# NEXT i
# OUTPUT "total: " + total
```

The hint students can ask for: The answer from `input()` is text, so make it a whole number first. FOR 1 TO steps includes the last one, so the range has to reach one past it. Add the 10 on before you print.

A solution

```
from bugbot import *
connect()
steps = int(input())
total = 0
for i in range(1, steps + 1):
    forward(50, distance=10)
    total = total + 10
    print(f"step {i}: {total}")
print("total:", total)
```

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