WORKSHEET

A15.4 Pseudocode in the exam languages

Exam preparation · A level · OCR H446 2.2.1, AQA 7517 4.4.1.2, Eduqas A500QS 1.8 · about 45 min

BugBotLab

Name _____ Class _____ Date _____

What this lesson is about

OCR's exam reference language and AQA's pseudo-code side by side with Python, the traps in translating, and a stack class that brings the robot home.

- The two languages side by side
- The traps in translating
- Reading a class in the exam reference language

Questions

5 marks in all

1. How many times does the AQA pseudo-code loop FOR i ← 1 TO 10 run? [1 mark]

- ☐ **A** 10
- ☐ **B** 9
- ☐ **C** 11
- ☐ **D** It depends on the language

2. Which Python matches the OCR loop do ... until reading < 30? [1 mark]

- ☐ **A** while True: with the body, then if reading < 30: break
- ☐ **B** while reading < 30: with the body
- ☐ **C** for reading in range(30): with the body
- ☐ **D** if reading < 30: with the body

3. In AQA pseudo-code, what does x ← 3 do? [1 mark]

- ☐ **A** Assigns 3 to x
- ☐ **B** Checks whether x equals 3
- ☐ **C** Outputs 3
- ☐ **D** Moves x three places left

4. In OCR's exam reference language, what is the name of a class's constructor? [1 mark]

_____

_____

5. This is a stack from the exam reference language, translated. What does it print? [1 mark]

```
items = [None] * 5
top = -1
for move in ["forward", "right", "left"]:
    top = top + 1
    items[top] = move
while top != -1:
    print(items[top])
    top = top - 1
```

The task: there and back with a stack

Translate the `Stack` class above into Python, then use it to bring the robot home. Write `class Stack` with: - `__init__(self)` : an empty stack; - `push(self, item)` : puts `item` on the top; - `pop(self)` : removes and returns the top item; - `is_empty(self)` : returns `True` if there are no items, otherwise `False` . `route` is a list of moves in order. Each move is a tuple `(direction, cm)` : `direction` is one of `"forward"` , `"backward"` , `"left"` or `"right"` , and `cm` is a whole number of centimetres. Drive each move in order at speed 50, pushing it onto a stack after it is driven. The robot then reaches the charger: play a note and print `arrived` . To come home, loop `while not <stack>.is_empty()` : pop a move, print `undo <direction> <cm>` , and drive the **opposite** direction (forward and backward are opposites, as are left and right) the same distance at speed 50. After the loop print `home` . Do not reverse the list yourself: the stack must do it.

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

route = [("forward", 30), ("right", 20), ("forward", 15), ("left", 10)]
```

Plan your program here, then type it in and press Run.

Do it on the robot

www.bugbotlab.com/learn/a15-4-exam-pseudocode/

The simulator checks it and tells you when it passes. Nothing to install, no account.

Challenges

1. Translate `pop` exactly as the pseudocode has it, with an array of 20 and a `top` pointer, and check the robot still comes home.

2. Write `push` in AQA pseudo-code as a subroutine that takes the array, the pointer and the item, and returns the new pointer.
3. The pseudocode's `push` does not check whether the array is full. Add the check in pseudocode, and say what should happen.
