



F5.1 What an algorithm is

Algorithms · GCSE · OCR J277 2.1.2, AQA 8525 3.1.1, Edexcel 1CP2 1.2.1 ·
about 12 min

Name _____

Class _____

Date _____

What this lesson is about

Inputs, processes and outputs; what makes an algorithm precise; programs implement algorithms.

- An algorithm is not a program
- Inputs, processes and outputs
- What makes a good algorithm
- Many algorithms, one problem

Questions

5 marks in all

1. What is an algorithm? [1 mark]
- ☐ A A set of step-by-step instructions to solve a problem
 - ☐ B A program written in Python
 - ☐ C A type of computer
 - ☐ D A list of variables
2. "Drive towards the wall and stop when close." What is wrong with this as an algorithm? [1 mark]
- ☐ A It is ambiguous: towards and close are not precise
 - ☐ B It is too short
 - ☐ C It has no loop
 - ☐ D Nothing is wrong
3. A program asks for a number of sides, works out the angle and drives the shape. Which are inputs? [1 mark]
- Tick every answer that is true.
- ☐ A The number of sides
 - ☐ B The angle
 - ☐ C The shape driven
 - ☐ D The message printed
4. What does it mean for an algorithm to be finite? [1 mark]
- ☐ A It always finishes
 - ☐ B It uses few variables
 - ☐ C It only works on small numbers
 - ☐ D It has exactly one loop

5. In which forms can an algorithm be written?

[1 mark]

Tick every answer that is true.

- ☐ A A flowchart
- ☐ B Pseudocode
- ☐ C Program code
- ☐ D A trace table

The task: any shape from an answer

Turn the shape algorithm into a program. Ask `How many sides?` , work out the corner angle, drive the shape with 20 cm sides, and print `drew a shape with <n> sides` . The task answers 5 . Work the angle out; do not type it.

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

# 1. Ask how many sides.
# 2. Work out the angle.
# 3. Repeat for each side: drive 20 cm, turn.
# 4. Say how many sides were drawn.
```

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



Do it on the robot

www.bugbotlab.com/learn/f5-1-what-an-algorithm-is/

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

Challenges

1. Write an algorithm in plain English for making a cup of tea, precise enough for a robot. Swap with a partner and find the ambiguous steps.
2. Write the inputs, processes and outputs for a program that tells you whether a number is even.
3. Write an algorithm, then a program, that finds out whether a number is prime by trying divisors only up to its square root.