



F5.6 Binary search

Algorithms · GCSE · OCR J277 2.1.3, AQA 8525 3.1.3, Edexcel 1CP2 1.2.6 ·
about 15 min

Name _____

Class _____

Date _____

What this lesson is about

Halving a sorted list, and why it needs sorted data.

- The algorithm
- Hear it halve
- Why it needs sorted data

Questions

5 marks in all

- What must be true of a list before binary search can be used? [1 mark]
 - ☐ A It must be sorted
 - ☐ B It must be short
 - ☐ C It must hold numbers only
 - ☐ D It must have an even number of items
- Binary search on a sorted list of 1,000 items. What is the most checks it needs? [1 mark]
- Binary search looks for 23 in [2, 5, 8, 12, 16, 23, 38, 56, 72, 91], checking index 4 (16) first. What happens next? [1 mark]
 - ☐ A It discards 16 and everything before it, and searches the upper half
 - ☐ B It discards the upper half
 - ☐ C It checks index 5 next, then 6, then 7
 - ☐ D It starts again at index 0
- Why might a programmer use linear search instead of binary search? [1 mark]
 - ☐ A The data is not sorted
 - ☐ B Linear search is always faster
 - ☐ C Binary search cannot find numbers
 - ☐ D Binary search needs more memory
- What does this program print? [1 mark]

```
low, high = 0, 9
mid = (low + high) // 2
print(mid)
```

The task: binary search the markers

The task asks `Which marker?` and answers `56`. Using **binary search** (not `in` or `.index`), find it in the sorted list `[2, 5, 8, 12, 16, 23, 38, 56, 72, 91]`. Beep once for every check, and print `found 56 at index <i> after <n> checks`.

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

ids = [2, 5, 8, 12, 16, 23, 38, 56, 72, 91]
target = int(input("Which marker? "))
```

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



Do it on the robot

www.bugbotlab.com/learn/f5-6-binary-search/

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

Challenges

1. Play the number guessing game with a partner using binary search. Is seven guesses always enough for 1 to 100?
2. Change `binary_search` so it returns how many checks it took, and test it on every item of a list.
3. What happens if the list has repeated items? Does binary search still find one of them?