



F8.2 Binary and denary

Data representation · GCSE · OCR J277 1.2.4, AQA 8525 3.3.1, Edexcel
1CP2 2.1.3 · about 15 min

Name _____

Class _____

Date _____

What this lesson is about

Converting 8-bit numbers both ways, and hearing them on the buzzer.

- Place values
- Denary to binary
- Binary to denary in Python
- Hear a number

Questions

6 marks in all

1. What is 01001101 in denary? [1 mark]

2. Write 200 as an 8-bit binary number. [1 mark]

3. What is the largest number 8 bits can hold? [1 mark]

4. Write 5 as an 8-bit binary number. [1 mark]

5. What does this program print? [1 mark]

```
total = 0
value = 1
for bit in reversed("1010"):
    if bit == "1":
        total = total + value
    value = value * 2
print(total)
```

6. What is the place value of the leftmost bit in an 8-bit binary number? [1 mark]

- ☐ A 128
- ☐ B 256
- ☐ C 100
- ☐ D 8

The task: play in binary

Write `to_binary(n)` yourself, without `bin`, `format` or `int(..., 2)`, that returns an 8-character string of 1s and 0s. Use it to print the binary of 77, 5 and 200, one per line, and then play the bits of 77: a note of 880 Hz for each 1 and 440 Hz for each 0, each for 0.2 seconds.

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

def to_binary(n):
    return ""
```

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



Do it on the robot

www.bugbotlab.com/learn/f8-2-binary-and-denary/

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

Challenges

1. How many bits are needed for the number 1,000? Write a loop that doubles until it passes 1,000.
2. Convert your age to binary by hand, then check with your function.
3. Write `to_denary` with the dividing-by-2 idea turned round: start from the left, and for each bit double the total and add the bit.