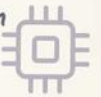




counting in four bases, and the complement trick that turns subtraction into addition



A base is a promise about place value *

- in any base b , the digit sitting k places left of the point is worth b to the power k , and the digits allowed run from 0 up to b minus 1
- so base 2 allows only 0 and 1, base 8 allows 0 to 7, base 10 allows 0 to 9, and base 16 allows 0 to 9 then A B C D E F for ten to fifteen
- the same written symbols mean different numbers in different bases: 10 is two in binary, eight in octal, sixteen in hexadecimal
- write the base as a subscript whenever there is any doubt, because a bare 110 is ambiguous and an examiner will read it as decimal



place value in base 2: each cell carries the weight written above it, so this pattern is 8 plus 2 plus 1

The four bases the paper uses

Binary, base 2

digits 0 and 1, the machine's own representation, every other base here is a convenience for humans reading it

Octal, base 8

digits 0 to 7, one octal digit stands for exactly three bits

Decimal, base 10

digits 0 to 9, the base every answer is finally checked against

Hexadecimal, base 16

digits 0 to 9 then A to F, one hex digit stands for exactly four bits

Reading any base back into decimal

positional expansion

worked, binary

worked, hexadecimal

multiply each digit by its own weight and add, whatever the base

$$N = d_n b^n + d_{n-1} b^{n-1} + \dots + d_1 b^1 + d_0 b^0$$

$$1011_2 = 1(8) + 0(4) + 1(2) + 1(1) = 11_{10}$$

$$2F_{16} = 2(16) + 15(1) = 47_{10}$$

