Exercise walk-through

4.2
Counting Systems

GAC Mathematics

Sets, relations and functions (2)

- A **relation** pairs elements of two sets. A **function** is a **relation** in which each input has exactly one output. "Each student $\to$ their birthday" is a **function** (two students may share a birthday); "each birthday $\to$ the students born on it" is not, because one input can have several outputs.
- A **Venn diagram** , introduced by the English logician John Venn in 1880, turns a set problem into a picture. Check that the regions add up to the total.

$$N_{neither} = |U| - |F \cup G| = 30 - 26 = 4$$

Counting systems

- A **number base** b uses the digits 0 to $b-1$, and each place is worth a power of b. **Decimal** uses ten, **binary** two, **hexadecimal** sixteen (digits 0–9 then A–F).
- Every digit's value is its **place value**: in binary the places are 1, 2, 4, 8, 16 and so on.
- **Hexadecimal** is shorthand for binary: one hex digit is exactly four bits. Web colours use it: ‘#FF8000‘ means red 255, green 128, blue 0, a bright orange.

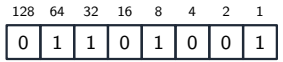
$$V = \sum d_i 2^i$$

Graphs and networks

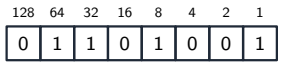
- A **graph** is a set of **vertices** joined by **edges**. It models anything with connections: roads, friendships, dependencies.
- The **degree** of a vertex is the number of **edges** that meet it. Every edge has two ends, so the degrees always add up to twice the number of edges. This is the fact behind Euler's Königsberg proof.
- A **tree** is a connected **graph** with no cycles; a **tree** with n vertices has $n - 1$ edges. A family tree and a computer's folder structure are trees.

Exercises 2.1·2.2

2.1 Convert 10110_2 to decimal. [2]

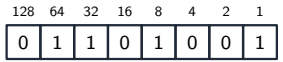


2.2 Write decimal 13 as an unsigned four-bit binary pattern. [2]



Exercises 2.1·2.2

2.1 Convert 10110_2 to decimal. [2]



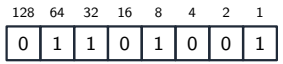
2.1

Use binary place values.

$$V = 1(16) + 0(8) + 1(4) + 1(2) + 0(1) = 22$$

Decimal 22.

2.2 Write decimal 13 as an unsigned four-bit binary pattern. [2]



2.2

Decompose 13 into weights 8, 4 and 1. Four-bit result 1101.

Exercises 2.3·2.4

2.3 Convert $2F_{16}$ to decimal and to an eight-bit binary pattern. [3]

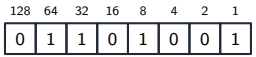
128	64	32	16	8	4	2	1
0	1	1	0	1	0	0	1

2.4 Calculate $1011_2 + 0110_2$ as an unrestricted binary numeral. Then state whether it fits in four unsigned bits. [3]

128	64	32	16	8	4	2	1
0	1	1	0	1	0	0	1

Exercises 2.3·2.4

2.3 Convert $2F_{16}$ to decimal and to an eight-bit binary pattern. [3]



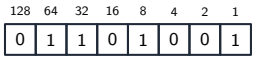
2.3

F represents 15.

$$V = 2(16) + 15 = 47$$

Decimal 47. Each hexadecimal digit supplies four bits, so binary pattern 00101111.

2.4 Calculate $1011_2 + 0110_2$ as an unrestricted binary numeral. Then state whether it fits in four unsigned bits. [3]

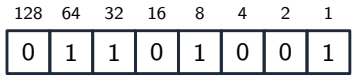


2.4

Column addition gives binary 10001. Carry method earns M1 and correct result A1. It represents 17 and exceeds the four-bit unsigned maximum 15, so it does not fit.

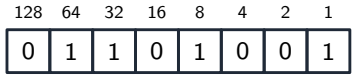
Exercise 3.1

3.1 (a) Write decimal 45 as an unsigned eight-bit pattern and hexadecimal numeral. (b) Subtract 00101101_2 from 01000000_2 as nonnegative integers. Give the result as eight bits and decimal. [6]



Exercise 3.1

3.1 (a) Write decimal 45 as an unsigned eight-bit pattern and hexadecimal numeral. (b) Subtract 00101101_2 from 01000000_2 as nonnegative integers. Give the result as eight bits and decimal. [6]



3.1

(a) Decompose 45 as $32 + 8 + 4 + 1$. Eight-bit pattern 00101101. Hexadecimal 2D.

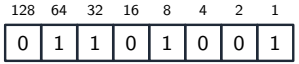
(b) Subtract values or use binary borrowing.

$$V = 64 - 45 = 19$$

Decimal result 19. Eight-bit pattern 00010011.

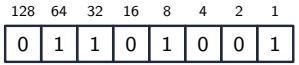
Exercise 3.2

3.2 An unsigned eight-bit counter starts at 11111010_2. A calculation adds decimal 10. (a) Find the initial decimal value and the unrestricted sum. (b) Write the unrestricted sum in binary. (c) Decide whether it fits the counter. (d) Under a separately specified wraparound rule that keeps only the lowest eight bits, find the stored pattern and decimal value. Explain why the stored value differs from the mathematical sum. [8]



Exercise 3.2

3.2 An unsigned eight-bit counter starts at 11111010_2. A calculation adds decimal 10. (a) Find the initial decimal value and the unrestricted sum. (b) Write the unrestricted sum in binary. (c) Decide whether it fits the counter. (d) Under a separately specified wraparound rule that keeps only the lowest eight bits, find the stored pattern and decimal value. Explain why the stored value differs from the mathematical sum. [8]



3.2

(a) Initial value is $V = 128+64+32+16+8+2 = 250$. Mathematical sum $S = 250+10 = 260$.

(b) Use place weights to represent 260. Binary numeral 100000100.

(c) Eight unsigned bits can hold at most $2^8 - 1 = 255$, so overflow occurs.

(d) Keep the low eight bits under the explicitly stated rule. Stored pattern 00000100, value 4. The high 256-weight bit is discarded, so this stored value is the remainder modulo 256 rather than the mathematical sum 260.