1.4 DATA TYPES & STRUCTURES · 1.4.1

Primitive types, binary & hexadecimal

Data types and base conversions — guaranteed marks. Spec 1.4.1(a)(b)(e)(f).

01 Primitive types

Integer	Whole number (42, –7).
Real	Has a fractional part (3.14). Int ÷ int → real.
Character	One symbol ('A').
String	Sequence of characters ("hello").
Boolean	True / False.
Bits	**n bits → 2^n values** (8 bits = 256, 0–255).

02 Binary ↔ denary

Place vals	`128 64 32 16 8 4 2 1`
Bin→den	Add place values where bit = 1.
Den→bin	Subtract largest value that fits, top-down.
Example	`01001101` = 64+8+4+1 = **77**
Why binary	Two reliable states (on/off) → robust.

03 Hexadecimal (base 16)

Digits	0–9 then **A=10 B=11 C=12 D=13 E=14 F=15**.
Key fact	**1 hex digit = 4 bits (a nibble)**; 1 byte = 2 hex digits.
Bin→hex	Group into nibbles from the right: `1001 0101` → **95**.
Hex→bin	Expand each digit to 4 bits: `F` → **1111**.
Hex→den	(first×16)+(second): `5A` = 80+10 = **90**.
Why hex	Easier to read, type & remember than binary.

FINAL PASS BEFORE THE EXAM

Rapid exam tips

Six slips on conversion questions.

01

Left bit of a byte = **128**, right bit = **1**. Line up place values first.

02

Hex goes 0–9 then **A–F** (10–15) — don't write "10, 11".

03

Group binary into nibbles of 4 **from the right**; pad with leading 0s.

04

Label bases ($_{2}$ $_{10}$ $_{16}$ / 0x) so you don't mix them up.

05

n bits → 2^n values. 8 bits = 256 (0–255).

06

Pick the data type from the value: price→real, count→integer, yes/no→Boolean.