

1.4 DATA TYPES & STRUCTURES · 1.4.1

Character sets: ASCII & Unicode — Mark scheme

25 marks · spec 1.4.1(j)

AO key: AO1 = knowledge · AO2 = application. Accept any valid alternative; do not award the same point twice.

Q	ANSWER	AO	MARKS
1(a)	A character set / standard that assigns a numeric (binary) code to each character (1); using 7 bits, covering English letters, digits, punctuation and control characters (1). (2)	AO1	2
1(b)	7 bits. (1)	AO1	1
1(c)	128 (= 2 ⁷). (1)	AO1	1
1(d)	An agreed set of characters each given a unique (binary) code. (1)	AO1	1

Q	ANSWER	AO	MARKS
2(a)	67 (A=65, so C=65+2). (1)	AO2	1
2(b)	C = 01000011 (1); A = 01000001 (1); B = 01000010 (1). (3)	AO2	3
2(c)	'7' is stored as its character code (55), not the integer value (1); the character set maps the symbol to a code, which is different from the number it depicts (1). (2)	AO2	2

Q	ANSWER	AO	MARKS
3(a)	ASCII's 128 codes only cover English (1); Unicode was needed to represent the characters of many other languages (and symbols/emoji) (1). (2)	AO1	2
3(b)	Any one: both are character sets mapping characters to binary codes · Unicode's first 128 codes are the same as ASCII. (1)	AO1	1
3(c)	1 mark each, max 2: Unicode uses more bits per character · Unicode represents far more characters · Unicode covers many languages (ASCII English only) · Unicode uses more storage. (2)	AO1	2
3(d)	It uses more memory/storage per character. (1)	AO1	1

Q	ANSWER	AO	MARKS
4(a)	Unicode. (1)	AO2	1
4(b)	2 reasons developed, 2 marks each (point + application), max 4: ASCII can't represent non-English characters (1) so users writing in other languages couldn't be supported (1); Unicode covers (almost) all languages and emoji (1) which an international messaging app needs (1). (4)	AO2	4
4(c)	Unicode uses more bits per character than ASCII (1); so each message takes more storage than the same text in ASCII (1); over millions of messages this significantly increases storage needs (1). (3)	AO2	3

Total for paper: 25 marks