

1.2 SOFTWARE & SOFTWARE DEVELOPMENT · 1.2.4 · FULL A-LEVEL

Assembly language, the LMC & addressing modes — Mark scheme

36 marks · spec 1.2.4(c)(d)

AO key: AO1 = knowledge & understanding · AO2 = application (tracing) · AO3 = design/program & reasoned judgements. Accept any valid alternative; do not award the same point twice.

Q	ANSWER	AO	MARKS
1	1 mark per correct cell, max 5: • INP (1) · STA addr = store the accumulator's value at addr (1) · BRZ (1) · ADD addr = add the value at addr to the accumulator (1) · HLT (1)	AO1	5

Q	ANSWER	AO	MARKS
2(a)	9 (1)	AO2	1
2(b)	7 (1)	AO2	1
2(c)	It compares the two inputs (1) and outputs the larger of the two numbers (1). (2)	AO2	2
2(d)	Any one of: INP / LDA / SUB (or STA's source is ACC — accept LDA first, SUB first, INP). (1) The instruction must place a (new) value in the accumulator.	AO1	1
2(e)	BRP bigger. (1)	AO1	1

Q	ANSWER (INDICATIVE)	AO	MARKS
3(a)	<pre> INP STA num1 INP ADD num1 OUT HLT num1 DAT </pre> 1 mark each, max 4: both inputs (1); first value stored with STA (1); ADD used to form the sum (1); OUT then HLT, with a DAT for the named location (1).	AO3	4
3(b)	Reserves a memory location labelled "total" (1) holding the initial value 0 (1). (2)	AO1	2

Q	ANSWER	AO	MARKS
4	2 differences × (both sides), max 4, e.g.: • assembly maps ~1:1 to machine code; one high-level line becomes many instructions (1+1) • assembly is hardware-specific; high-level code is portable (1+1) • assembly is harder/slower to write and read; high-level is closer to English (1+1)	AO1	4

Q	ANSWER	AO	MARKS
5(a)	1 mark each, max 2: faster execution / smaller code, less memory / direct control of the hardware (e.g. embedded systems, drivers). (2)	AO1	2
5(b)	An assembler. (1)	AO1	1

Q	LEVELS-OF-RESPONSE MARK SCHEME	AO	MARKS										
6	Mark using the levels descriptors below. AO1 (knowledge of the four modes), AO2 (how the operand is interpreted, with examples), AO3 (reasoned discussion of why different modes are useful). <table><tr><th>LEVEL</th><th>DESCRIPTOR</th></tr><tr><td>Level 3 (9–12)</td><td>Accurately explains all four modes, including how the operand is interpreted in each, with clear reasoning about why different modes are useful (speed, address range, arrays). Well structured with accurate terminology.</td></tr><tr><td>Level 2 (5–8)</td><td>Explains most modes correctly with some reasoning about their usefulness; minor confusion or omissions.</td></tr><tr><td>Level 1 (1–4)</td><td>Basic points about one or two modes with little explanation of how operands are interpreted or why modes differ.</td></tr><tr><td>0</td><td>Nothing creditworthy.</td></tr></table> Indicative content (credit any valid point): • Immediate: the operand is the value itself — no memory access, so fast; good for constants. • Direct: the operand is the address of the value — simple, one memory access; limited by operand width. • Indirect: the operand is the address of a location holding the value's address — allows a much larger address space than the operand field, supports pointers. • Indexed: address = operand + index register — stepping the index register walks through an array without changing the instruction. • Usefulness: trade-offs between speed, reachable memory and flexible data structures explain why all four exist.	LEVEL	DESCRIPTOR	Level 3 (9–12)	Accurately explains all four modes, including how the operand is interpreted in each, with clear reasoning about why different modes are useful (speed, address range, arrays). Well structured with accurate terminology.	Level 2 (5–8)	Explains most modes correctly with some reasoning about their usefulness; minor confusion or omissions.	Level 1 (1–4)	Basic points about one or two modes with little explanation of how operands are interpreted or why modes differ.	0	Nothing creditworthy.	AO1 ×4 AO2 ×4 AO3 ×4	12
LEVEL	DESCRIPTOR												
Level 3 (9–12)	Accurately explains all four modes, including how the operand is interpreted in each, with clear reasoning about why different modes are useful (speed, address range, arrays). Well structured with accurate terminology.												
Level 2 (5–8)	Explains most modes correctly with some reasoning about their usefulness; minor confusion or omissions.												
Level 1 (1–4)	Basic points about one or two modes with little explanation of how operands are interpreted or why modes differ.												
0	Nothing creditworthy.												

Total for paper: 36 marks