MODULE M PROGRAMMING · M21

File handling: open, read, write and close: mark scheme

20 marks · Cambridge 0478 syllabus (exams 2026-2028) 8.3.1 (purpose of a file), 8.3.2 (open, close, read and write files), with 8.1.4(c), 7.8 and 7.9

How to use. One mark per creditworthy point up to the maximum shown. **/** separates alternative words; **//** separates alternative answers. Words in (brackets) are not needed for the mark. Credit any other valid answer that shows the same understanding. Written algorithms must be in pseudocode; ignore minor syntax slips that do not change the logic. File names may be written with or without quotation marks.

QUESTION	ANSWER	MARKS	GUIDANCE
1 0478 8.3.1	Any two from: The value in the variable is lost when the program stops // when the machine is switched off Data in a file is stored permanently // is not lost when the program stops // the file is kept in non-volatile / secondary storage The count can be read back / loaded into the program the next time it runs // the count can carry on the next day The data in the file can be used by another program // can be accessed later (e.g. to check the day's sales)	**2**	Max 2. Credit any other valid reason. Do not credit "the file is safer / more secure" or "the file is faster".
2 0478 8.3.2	One mark for each correct row: `OPENFILE ... FOR WRITE` adds new data after the existing data: False A file should be opened in only one mode at a time: True `READFILE "Stock.txt", Item` reads a data item and assigns it to `Item`: True `CLOSEFILE "Stock.txt"` deletes the file: False	**4**	Row 1: opening FOR WRITE creates a new file and any existing data is lost. Row 4: CLOSEFILE closes the file when it is no longer needed; the file and its data are kept. A row with two ticks scores 0 for that row.
3(a) 0478 8.3.2 0478 8.1.4(c)	Total rainfall: 19	**1**	12 + 0 + 7 = 19. Allow 19 alone. Ignore quotation marks and the spacing between the message and the number. Do not credit `Total`.
3(b) 0478 8.3.2	READ	**1**	Ignore case. Do not credit `READFILE` or `OPENFILE` (those are commands, not the mode).

QUESTION	ANSWER	MARKS	GUIDANCE
3(c) 0478 8.3.2	The file contains only the total // only 19 // one line holding 19 ... because opening the file FOR WRITE creates a new file // the existing data / the three readings 12, 0 and 7 are lost / overwritten	**2**	Second mark: the explanation must link the loss of the old data to opening the file FOR WRITE (or to writing to the file). An answer saying the file holds 12, 0, 7 and 19 scores 0.
4 0478 8.3.2 0478 7.8	One mark for each error identified and one mark for its correction: Line 04 ... `OPENFILE "Members.txt" FOR WRITE` Line 05 ... `WRITEFILE "Members.txt", NewName`	**4**	Each correction mark depends on the matching line number. Allow a precise description (e.g. "change READ to WRITE"). Do not credit lines 01, 02, 03 or 06 as errors.

5
0478 8.3.2
0478 8.1.4(c)
0478 7.9
	One mark per bullet point. Max 6:
	Declares a STRING variable for the line of text (and an INTEGER loop counter if one is used)
	`OPENFILE "Tools.txt" FOR READ`
	`OPENFILE "Backup.txt" FOR WRITE` (before the loop)
	Loop that repeats five times // `FOR Count ← 1 TO 5` ... `NEXT Count`
	`READFILE "Tools.txt", LineOfText` inside the loop
	... `WRITEFILE "Backup.txt", LineOfText` inside the loop, using the same variable
	Both files closed with `CLOSEFILE` after the loop
	Message output after the loop, e.g. `OUTPUT "Copy complete"`
	Example answer:
	`DECLARE LineOfText : STRING`
	`DECLARE Count : INTEGER`
	`OPENFILE "Tools.txt" FOR READ`
	`OPENFILE "Backup.txt" FOR WRITE`
	`FOR Count ← 1 TO 5`
	`  READFILE "Tools.txt", LineOfText`
	`  WRITEFILE "Backup.txt", LineOfText`
	`NEXT Count`
	`CLOSEFILE "Tools.txt"`
	`CLOSEFILE "Backup.txt"`
	`OUTPUT "Copy complete"`
		6	Accept a WHILE or REPEAT loop with a counter that runs exactly five times. Accept five separate READFILE / WRITEFILE pairs instead of a loop: award the loop mark only for a loop. Do not award the `FOR WRITE` mark if `Backup.txt` is opened FOR WRITE inside the loop (each opening would lose the line written before). Do not award the READFILE or WRITEFILE marks if the file and variable are the wrong way round (penalise once only). Program code (instead of pseudocode) scores 0.

Total: 20 marks