# A-Level Computer Science

**Common Mistakes ·**

*Two things cost this subject marks: writing everyday English where the syllabus has a technical term, and answering in general terms where the question set a scenario. Everything below is from Cambridge's own examiner reports.*

## Across every paper

✗ Answering in everyday words – 'the computer remembers it', 'it makes things faster'.

✓ **Use the syllabus term: 'the value is stored in the accumulator', 'cache reduces the average access time'.**

Named in six separate reports. The technical words have defined meanings and marks are awarded for using them.

ER 6 sessions P1

✗ Describing a benefit in general terms when the question named a company, a system or a file.

✓ **Answer inside the scenario: 'the supermarket's stock file is updated as each item is scanned'.**

Examiners state that answers must refer to the given context to receive maximum credit – 9618 emphasises application over recall.

ER 3 sessions P1

✗ Answering an EXPLAIN question with the detail you would give for DESCRIBE.

✓ **EXPLAIN needs a reason: not just what happens, but why it has that effect.**

The most repeated command-word point in these reports, raised in seven sessions.

ER 7 sessions P1

✗ Making the same point twice in different words to fill two marks.

✓ **Make each point sufficiently different that it stands alone.**

Examiners state plainly that two marks are not awarded for two statements that mean the same thing.

ER 2 sessions P1

✗ Giving 'memory management' when the question said 'describe two OTHER operating system tasks'.

✓ **Read for what the question excludes as well as what it asks.**

Examiners give this exact example: specific answers are sometimes ruled out by the stem.

ER 2 sessions P1

✗ Leaving a pseudocode algorithm question blank because the whole algorithm looks hard.

✓ **Write the part you can – the loop, the condition, the assignment.**

Examiners state these questions are at least partially accessible to most candidates. Nothing can be awarded for a blank.

ER 6 sessions P1

- Prepare with the published pseudocode guide, especially for user-defined types, file handling and algorithm construction. Examiners recommend it in eight separate reports.

ER 8 sessions P1

## 1 · Information representation

✗ Explaining an overflow by saying ‘the answer was greater than 255’ or ‘the answer has 9 bits’.

✓ **State the word size the question gave, then say the result cannot be represented in it.**

Examiners record candidates assuming 8-bit values when the question specified otherwise, so the reasoning was right and the answer wrong.

EX P1

✗ Drawing two lines from one row of a truth table to be safe.

✓ **Draw exactly one line per row.**

Examiners note candidates drawing more than one line where the question asked for one – the extra line loses the mark.

EX P1

## 8 · Databases

✗ Answering a relationship question with ‘one-to-many’.

✓ **Name the tables: one SHOW to many PERFORMANCEs.**

Examiners accepted the named form and rejected the bare cardinality – without the table names it does not answer the question.

EX P1

## 10 · Data Types and Structures

✗ Saying an array makes ‘the data easier to access’.

✓ **Name the structure the question named: an array OF RECORDS lets each field be accessed by name within one indexed element.**

Examiners record that answers referring only to ‘an array’ were not awarded, because the question asked about an array of records.

“ ”

EX P1

✗ Creating an array called TheTree when the question wanted a class instance.

✓ **Declare an instance of the class: ‘TheTree = NEW BinaryTree()‘.**

A named error. An array is not an object, so none of the methods that follow can be called on it.

,

EX P4

✗ Initialising a node’s left and right pointers by assigning parameter values to them.

✓ **Initialise both pointers to −1 (or null) and take only the data as a parameter.**

Examiners list taking three parameters instead of one, and assigning to the pointers, as the common constructor errors.

−1

EX P4

✗ Passing an integer to a constructor that expects a Node.

✓ **Pass the element itself, so no new Node has to be declared inside.**

Examiners note this forced candidates to declare extra Node objects, which then broke the tree structure.

EX P4

## 11 · Programming

✗ Writing a Python method without ‘self‘ in the parameter list.

✓ **‘def InsertNode(self, NewNode):‘ – every method takes ‘self‘ first.**

Listed as a common error in two separate questions. The code will not run, so the marks for it cannot be awarded.

Python self EX P4

✗ Assigning constructor parameters to attributes and also declaring them as parameters again.

✓ **Take the parameter once, assign it to the attribute once.**

Named among the constructor errors; the duplication produces attributes that are never the ones used.

EX P4

✗ Calling a function and not storing what it returns.

✓ **Assign the result: ‘SortedArray ← BubbleSort(MyArray, 7)‘.**

Listed repeatedly – the function runs, the answer is computed, and it is thrown away.

EX P4

✗ Calling a search function with the original unsorted array.

✓ **Sort first, then pass the sorted array, and show that you did.**

Examiners require the sorted array to be used AND to be visibly used; a binary search on unsorted data is meaningless.

EX P4

✗ Calling a function with the array length one out – 6 for a seven-element array, or the last index instead of the length.

✓ **Decide whether the parameter is a LENGTH or an INDEX, and check the last element is included.**

Both forms are listed as common errors, and each silently drops the final element.

EX P4

✗ Using the input as a filename directly.

✓ **Append the extension the question specified: ‘FileName ← Choice & “.txt”‘.**

A named error: the file then never opens, so nothing downstream can score.

EX P4

✗ Closing a file inside the loop that reads it.

✓ **Close it once, after the loop has finished.**

Examiners describe candidates closing the file and then attempting to read more, or having to reopen it.

EX P4

## 19 · Computational thinking and Problem-solving

✗ Writing an iterative function for a question that asked for a recursive one.

✓ **The function must call itself, and the call must reduce the problem.**

Examiners list writing an iterative function with no recursive call, and including a loop AS WELL AS the recursion, as common errors.

EX P4

✗ Making a recursive call without reducing the parameter.

✓ **Subtract one: 'RETURN Total(NumberElements - 1)'.**

Named as an inaccurate recursive call. Without the reduction the recursion never terminates.

EX P4

✗ Overwriting a parameter inside the function before it has been used.

✓ **Read the parameter first, then reuse the variable if you must.**

Listed as a common error in two separate recursion questions – the original value is gone before the recursive call needs it.

EX P4

✗ Converting a recursive routine to an iterative one by renaming the identifier.

✓ **Remove the recursive call AND add the loop that replaces it.**

Examiners describe both halves being left undone: the call remained, or it went and nothing took its place.

EX P4

✗ Printing every element when the question asked for the nodes.

✓ **Output what was asked: for each node, the left pointer, the data and the right pointer.**

Both faults are listed together – outputting too much, and outputting only the data.

EX P4

## The practical paper – evidence and screenshots

✗ Pasting part of a program into the evidence document.

✓ **Copy ALL of it, including every closing bracket and END statement, and check the indentation survived.**

Examiners require the complete code in the answer space, and note that Python candidates in particular must check their indentation is intact.

ER 2 sessions P4

✗ Showing only the output and not the input that produced it.

✓ **Screenshot the whole run – the prompt, what was typed, and every line of output.**

Listed repeatedly: outputs cut off, only the first few results shown, or the input missing entirely.

EX P4

✗ Outputting the results but not the value the question also asked for.

✓ **Reread the question and check every requested item appears – the word AND the percentage, the count AND the list.**

Examiners list this pattern in four separate questions; the code was right and the output incomplete.

EX P4

✗ Putting an answer in the wrong space in the evidence document, or deleting spaces you did not use.

✓ **Leave every answer space in place and put each piece of evidence in its own.**

Examiners state this explicitly. Evidence in the wrong space cannot be marked as if it were in the right one.

ER 2 sessions P4

■ A screenshot must be legible. Examiners record evidence that could not be read, which is the same as evidence that was not provided.

EX P4