

*9618 is marked on precision of expression more than on programming ability. Most lost marks come from describing a mechanism loosely, or from answering with an example where a definition was wanted. 9618 重在表述精确：机制要写清楚，定义题不能只举例子。*

## Terminology

---

Explaining an interrupt as ‘the computer stops’. It suspends the current process, saves its state to a stack, runs a service routine and RESTORES the state – the saving and restoring are the marks.

Describing virtual memory as ‘extra RAM’. It is secondary storage used as if it were RAM, with pages swapped in and out.

Confusing a compiler’s error report with an interpreter’s. A compiler lists all errors after the whole translation; an interpreter stops at the first.

Saying a hash function ‘encrypts’ a password. Hashing is one-way and cannot be reversed, which is precisely why it is used.

Using ‘data’ and ‘information’ interchangeably. Information is data that has been processed and given context.

## Data structures and algorithms

---

Describing a stack as ‘first in, first out’. A stack is LIFO; a queue is FIFO. Reversing them loses every subsequent mark.

Traversing a binary tree without saying which order. In-order, pre-order and post-order give different sequences and the question always specifies one.

Claiming binary search works on any list. It requires the data to be SORTED, and saying so is usually a mark.

Giving a Big-O answer without justification. State what n counts and why the operations grow that way.

Writing a linked-list operation without updating the pointers of BOTH neighbours, or without handling the empty-list case.

## Databases and networking

---

Choosing a primary key that could repeat, or forgetting that a composite key needs both attributes to be stated.

Describing normalisation by naming the normal form without saying what it removes – 2NF removes partial dependency, 3NF removes transitive dependency.

Saying TCP ‘sends data faster’. TCP is reliable and connection-oriented; UDP is faster because it does not guarantee delivery.

Confusing the roles in the client-server model, or describing peer-to-peer as having no servers rather than every peer acting as both.

## Programming and answers

---

Answering a ‘describe an algorithm’ question with code when pseudocode or prose was asked for, or the reverse.

Writing a recursive routine with no base case, or with a base case that the recursion can never reach.

Giving one test datum where the question asked for normal, boundary and erroneous cases – three types, three marks.

Explaining object-oriented terms with an example instead of a definition: ‘a Dog is an Animal’ does not define inheritance.