

A-Level Computer Science

Name: _____ Score: _____

1. Abstraction means:

- ☐ keeping the essential features and ignoring irrelevant detail
- ☐ breaking a problem into smaller parts
- ☐ writing the program in machine code
- ☐ testing every possible input

2. An algorithm is "deterministic". This means:

- ☐ the same input always produces the same output
- ☐ it runs forever
- ☐ each step has several meanings
- ☐ it needs no input

3. In a flowchart, which shape represents a decision?

- ☐ a diamond
- ☐ a rounded rectangle
- ☐ a parallelogram
- ☐ a plain rectangle

4. Decomposition means:

- ☐ breaking a large problem into smaller sub-problems
- ☐ ignoring irrelevant detail
- ☐ encrypting the data
- ☐ running tasks in parallel

5. How does a WHILE loop differ from a REPEAT...UNTIL loop?

- ☐ WHILE tests the condition before each pass, so it may run zero times
- ☐ WHILE always runs at least once
- ☐ REPEAT tests before the first pass
- ☐ they are identical

6. A train map that keeps the stations and lines but drops the real geography is an example of abstraction.

- ☐ True
- ☐ False

7. By De Morgan's law, 'NOT (A AND B)' is the same as '(NOT A) OR (NOT B)'.

☐ True ☐ False

8. What is the value of '17 MOD 5'?

9. A FOR loop is count-controlled (it repeats a fixed number of times), while a WHILE loop is condition-controlled (it repeats until a condition changes).

☐ True ☐ False

10. Stepwise refinement is the technique of:

- ☐ starting with a high-level outline and expanding each step until it can be coded
- ☐ writing the whole program in one go
- ☐ testing every input
- ☐ translating to machine code

1. keeping the essential features and ignoring irrelevant detail

Abstraction simplifies a problem to its essentials. Breaking into parts is decomposition.

2. the same input always produces the same output

Deterministic = same input $\rightarrow$ same output every time. (Finite = the steps end; unambiguous = one meaning per step.)

3. a diamond

A diamond is a decision; rounded rectangle = start/stop, parallelogram = input/output, rectangle = process.

4. breaking a large problem into smaller sub-problems

Decomposition splits a big task into smaller, separately-solvable sub-tasks.

5. WHILE tests the condition before each pass, so it may run zero times

WHILE checks first (can run 0 times); REPEAT...UNTIL checks after, so it always runs at least once.

6. True

It keeps what matters (stations, connections) and discards the rest — exactly what abstraction does.

7. True

NOT distributes over the bracket and flips AND $\leftrightarrow$ OR; likewise NOT (A OR B) = (NOT A) AND (NOT B).

8. 2

MOD gives the remainder: $17 = 3 \times 5 + 2$, so '17 MOD 5 = 2'. ('17 DIV 5 = 3:')

9. True

Use FOR when you know how many passes; use WHILE/REPEAT when you loop until something becomes true.

10. starting with a high-level outline and expanding each step until it can be coded

You refine a high-level outline level by level, adding detail while keeping the structure.