

KK06 — Debugging and alpha testing techniques for checking solutions meet requirements and function correctly

Development & Evaluation · Comprehensive exam study guide

STUDY DESIGN — YOU MUST KNOW

- breakpoints
- commenting out code
- relevant test data
- test cases comparing expected and actual output in testing tables

Alpha testing is testing by the developer (internally) to confirm the solution meets requirements and functions correctly. Know the techniques and be able to build a **testing table** — a frequent SAC/exam skill.

Techniques

Technique	Purpose
Breakpoint	Pauses execution at a line to inspect variable values and locate a fault.
Commenting out code	Temporarily disabling lines to isolate which part causes a problem.
Relevant test data	Valid, boundary and invalid inputs chosen to test behaviour.
Testing table	Records input, expected and actual output, and pass/fail for each test.

Alpha testing

Alpha testing is performed **by the developer**, before release, to find defects and confirm requirements are met. It contrasts with beta testing (KK07), done by real users.

EXAM TIP

Strong test data = typical + boundary + invalid. A testing table must compare **expected vs actual** to show pass/fail.

Key definitions to memorise

Alpha testing	Internal testing carried out by the developer to check the solution meets requirements and functions correctly.
Commenting out code	Temporarily disabling lines of code to isolate the source of a fault.
Testing table	A table comparing expected and actual output for each test case.

Exam technique

EXAM TIP

- Distinguish alpha (developer) from beta (real users) testing — a common question.
- Always include boundary and invalid test data, not just typical values.

COMMON MISTAKES

- Confusing alpha and beta testing.
- Building a testing table without an 'expected vs actual' comparison.

Quick revision checklist

- I can describe the four debugging/alpha-testing techniques.
- I can construct relevant test data and a testing table.
- I can distinguish alpha from beta testing.

Exam practice — questions & model answers

Q1. Explain how commenting out code helps locate a fault.

Explain · 2 marks

Model answer

- Disabling sections of code temporarily (1)
- lets the developer see whether the problem disappears, isolating the part responsible (1).

Q2. Construct a testing table with one typical, one boundary and one invalid test for a field accepting ages 0–120.

Construct · 3 marks

Model answer

- Typical: input 30 → expected accepted (1).
- Boundary: input 0 or 120 → expected accepted (1).
- Invalid: input –5 or 130 → expected rejected (1).