Exercise walk-through

Identifying and Correcting Errors ▪ 1.4

eXercise

You must be able to

- distinguish a **logic error** 逻辑错误, a **syntax error** 语法错误, a runtime error, and an overflow error
- use **testing** 测试 with varied inputs, including **edge cases** 边界情况
- locate and fix a defect during **debugging** 调试
- verify the corrected program produces the intended output

Method — Types of error

- **Syntax error** — breaks the language's grammar, so it will not run.
- **Logic error** — runs but gives the **wrong** result.
- **Runtime error** — crashes while running (e.g. dividing by zero).
- **Overflow error** — a value is too large for the bits available.

Method — Testing and debugging

Test with varied inputs, including **edge cases** (boundaries and unusual values). To find a bug, **hand-trace** the code or add temporary **print statements**, then re-test to verify the fix.

Practice 2.1 ▪ 2 marks

State the difference between a syntax error and a logic error.

Solution 2.1

Method: Types of error

Worked steps

a syntax error breaks the language's rules so the program will not run; a logic error runs but gives the wrong result B1 B1.

Practice 2.2 ▪ 1 mark

State what edge cases are used for in testing.

Solution 2.2

Method: Testing and debugging

Worked steps

to test the program at boundaries and unusual values where defects often hide B1.

Practice 2.3 ▪ 1 mark

Name one debugging strategy.

Solution 2.3

Worked steps

hand-tracing the code, or adding print statements / using a debugger (any one) B1.

Exam-style 3.1 ▪ 1 mark

A program runs but produces the wrong answer. This is a

A syntax error

B logic error

C overflow error

D case with no error

Solution 3.1

Method: Types of error

A syntax error

B logic error ✓

C overflow error

D case with no error

Worked steps

B.

Exam-style 3.2 ▪ 1 mark

Testing a program with unusual or boundary inputs uses

A edge cases

B comments

C variables

D loops

Solution 3.2

Method: Testing and debugging

A edge cases ✓

B comments

C variables

D loops

Worked steps

A.

Exam-style 3.3 ▪ 1 mark

A program crashes when it tries to divide by zero.
(a) Name the type of error.
(b) Name one way to find the bug.
(c) State what you should check after fixing it.

Solution 3.3

Method: Types of error

Worked steps

(a) a runtime error B1. (b) hand-trace or add print statements to find where it fails (any one) B1. (c) that it now produces the intended output for its requirements B1.