Your notes

GCSE Computer Science: revision notes

Robust programs and testing

Contents

1. Defensive design
2. Maintainability
3. Errors
4. Testing
5. Key terms
6. Quick questions
7. What to do next

Defensive design

- Anticipate misuse: think about the mistakes and mischief users might try, and plan for them.
- **Authentication** checks that users are who they say they are, for example with a username and password.
- **Validation** checks that data is sensible before it is used: a range check (between 1 and 100), a type check (a number), a length check (at least 8 characters), a presence check (not left blank) or a format check (matches a pattern, such as a postcode).
- **Verification** checks that data was entered correctly, for example by typing a new password twice.

Maintainability

- Make code easy to understand and update: use comments, meaningful names, indentation, subroutines and constants.

Errors

- A **syntax error** breaks the rules of the language, such as a missing bracket, so the program won't run.
- A **logic error** lets the program run, but it gives the wrong result, such as using < instead of <=.
- A runtime error happens while the program runs, such as dividing by zero.

Testing

- Iterative testing happens during development, as each part is written. Final (terminal) testing happens once the program is complete.

- Test data: **normal data** (typical values that should be accepted), **boundary data** (values at the edge of the valid range, and just outside it) and erroneous or invalid data (values that should be rejected).
- For valid ages 11 to 18: normal 14; boundary 11 and 18 (and 10 and 19 just outside); erroneous 'ten' or –5.
- A **test plan** lists each test, its test data, the expected result and the actual result.

Your notes

Key terms

Validation	Checking that data is sensible and allowed before it is used.
Verification	Checking that data has been entered correctly, such as by typing it twice.
Authentication	Checking that a user is who they claim to be.
Syntax error	An error that breaks the rules of the language, so the code won't run.
Logic error	An error that lets the program run but gives the wrong result.
Normal data	Typical, valid data that the program should accept.
Boundary data	Data at the edge of the valid range, or just outside it.
Erroneous data	Data that is wrong and should be rejected.
Test plan	A list of tests with the data, the expected result and the actual result.

Quick questions

Cover the answers and test yourself: remembering is what makes it stick.

1. What does a range check do?
2. Which check makes sure that a field is not left blank?
3. What is verification?
4. What is authentication?
5. A program is missing a closing bracket and won't run. What type of error is this?
6. A program runs, but it calculates the wrong total. What type of error is this?
7. Valid ages are 11 to 18. Which of these is boundary data?
8. Valid ages are 11 to 18. Which of these is erroneous data?

Answers

1. Checks that a value is between a lower and an upper limit
2. Presence check
3. Checking data has been entered correctly, such as typing it twice
4. Checking that a user is who they say they are
5. Syntax error
6. Logic error

7. 18
8. Twelve

What to do next

Practise 17 questions on this topic

Try 6 exam questions, with mark schemes

Read the lesson, one step at a time

Free revision for GCSE and KS3 at lycolearn.co.uk. Some notes are for one exam board only, and say so in brackets.

Your notes