Test design: Part II

Software Testing: IN3240 / IN4240
Summary

**Specification-based testing** *(black-box)*
- Equivalence partitioning | Boundary value analysis
- Decision table | State transition | Use case testing

**Structure-based testing** *(white-box)*
- Statement / Decision testing and coverage

**Experience-based testing**

**Choosing test technique**
Part I: Close-ended questions
Question 1

Which of the following would **structure-based test design techniques** be likely to be applied to?

1. Boundaries between mortgage interest rate bands
2. An invalid transition between two different arrears statuses
3. The business process flow for mortgage approval
4. Control flow of the program to calculate repayment

a. 2, 3 and 4  
b. 2 and 4  
c. 3 and 4  
d. 1, 2 and 3
Question 1: Clues

Which of the following would **structure-based test design techniques** be likely to be applied to?

Different types of testing
Which of the following would **structure-based test design techniques** be likely to be applied to?

**Purpose of structure-based techniques**

- Test coverage measurement
  - Assess the amount of testing performed by tests
  - Derived from specification-based technique to assess coverage

**Structural test design**

- Generate additional test cases
- Increase test coverage
Which of the following would structure-based test design techniques be likely to be applied to?

Characteristics of structure-based techniques

Target: Testing the structure of a system / component

White-box testing → What happens inside the box?

Can occur at any test level

Tends to be applied to component / integration level testing

Higher test levels → Business process testing

Control flow models

Support structural testing
Question 1: Answer

Which of the following would *structure-based test design techniques* be likely to be *applied* to?

1. Boundaries between mortgage interest rate bands
2. An invalid transition between two different arrears statuses
3. The *business process flow for mortgage approval*
4. Control flow of the program to calculate repayment

a. 2, 3 and 4
b. 2 and 4
c. 3 and 4
d. 1, 2 and 3
Use case testing is useful for which of the following?

1. Designing acceptance tests with users or customers
2. Making sure the mainstream business processes are tested
3. Finding defects in the interaction between components
4. Identifying the maximum and minimum values for every input field
5. Identifying the percentage of statements exercised by a set of tests

a. 1, 2 and 3
b. 2, 4 and 5
c. 1, 2 and 4
d. 3, 4 and 5
Use case testing is useful for which of the following?

Use case testing

Technique to identify test cases that exercise the whole system

Transaction by transaction basis from start to finish

Sequence of steps → Describes interactions between actor and system

Achieve a specific task / Produce something of value to the user

Defined in terms of the actor, not the system

Describes process flows through a system → Based on its actual use

Can uncover integration defects → Incorrect actions between components

Individual testing would not uncover these
Use case testing is useful for which of the following?

1. Designing acceptance tests with users or customers
2. Making sure the mainstream business processes are tested
3. Finding defects in the interaction between components
4. Identifying the maximum and minimum values for every input field
5. Identifying the percentage of statements exercised by a set of tests

a. 1, 2 and 3
b. 2, 4 and 5
c. 1, 2 and 4
d. 3, 4 and 5
Question 3

Which of the following statements about the relationship between statement and decision coverage is correct?

a. 100 % decision coverage is achieved if statement coverage is greater than 90 %
b. 100 % statement coverage is achieved if decision coverage is greater than 90 %
c. 100 % decision coverage always means 100 % statement coverage
d. 100 % statement coverage always means 100 % decision coverage
Question 3: Clues

Which of the following statements about the relationship between statement and decision coverage is correct?

Statement coverage

Code example (each line is a statement)

1. READ A
2. READ B
3. IF A > B THEN C = 0
4. ENDIF

To achieve 100% statement coverage:

How many test cases needed?
Question 3: Clues

Which of the following statements about the relationship between statement and decision coverage is correct?

**Achieving 100 % Statement coverage**

- Just one test case needed
- A must be greater than B
- Runs through all statements

Example test case

- \( A = 12 \)
- \( B = 10 \)
Question 3: Clues

Which of the following statements about the relationship between statement and decision coverage is correct?

Example II: Statement coverage

1. READ A
2. READ B
3. \( C = A + 2 \times B \)
4. IF \( C > 50 \) THEN
5. PRINT LARGE C
6. ENDIF
Question 3: Clues

Which of the following statements about the relationship between statement and decision coverage is correct?

Example II: Statement coverage

Test 1_1:
A = 2, B = 3

Test 1_2:
A = 0, B = 25

Test 1_3:
A = 47, B = 1

Which statements have we covered?
Question 3: Clues

Which of the following statements about the relationship between **statement** and **decision coverage** is correct?

Example II: Statement coverage

Test 1_1:

A = 2, B = 3 // C = 8

Test 1_2:

A = 0, B = 25

Test 1_3:

A = 47, B = 1

Which **statements** have we **covered**?
Question 3: Clues

Which of the following statements about the relationship between statement and decision coverage is correct?

Example II: Statement coverage

Test 1_1:
   A = 2, B = 3

Test 1_2:
   A = 0, B = 25 // C = 50

Test 1_3:
   A = 47, B = 1

Which statements have we covered?
Which of the following statements about the relationship between statement and decision coverage is correct?

Example II: Statement coverage

Test 1_1:
A = 2, B = 3

Test 1_2:
A = 0, B = 25

Test 1_3:
A = 47, B = 1 \(//\) C = 49

Which statements have we covered?
Question 3: Clues

Which of the following statements about the relationship between statement and decision coverage is correct?

Example II: Statement coverage

Have covered 5 out of 6 statements

Statement coverage = 83%

Need another test to reach 100%

Test 1_4:

A = 20, B = 25
Question 3: Clues

Which of the following statements about the relationship between statement and decision coverage is correct?

Example II: Statement coverage

Test 1_4:

A = 20, B = 25 // C = 70

Statement coverage = 100 %

In fact, only one test case needed
Question 3: Clues

Which of the following statements about the relationship between statement and decision coverage is correct?

Decision coverage

Each decision must have both a true and false outcome

Code example (each line is a statement)

1. READ A
2. READ B
3. IF A > B THEN C = 0
4. ENDIF

To achieve 100% decision coverage: How many test cases needed?
Question 3: Clues

Which of the following statements about the relationship between statement and decision coverage is correct?

**Decision coverage**

*One test required for 100 % statement coverage*

A = 12, B = 10

// All statements are exercised

Decision coverage requires:

Each condition must have True and False

Test case condition

A must be less than or equal to B
Question 3: Clues

Which of the following statements about the relationship between statement and decision coverage is correct?

Decision coverage

Additional test case

\[ A = 2, \quad B = 4 \]

// All decisions have been exercised

Have achieved 100% decision coverage
Question 3: Clues

Which of the following statements about the relationship between statement and decision coverage is correct?

Statement and Decision coverage

Statement coverage = \( \frac{\text{Number of statements exercised}}{\text{Total number of statements}} \) \times 100

Decision coverage = \( \frac{\text{Number of decision outcomes exercised}}{\text{Total number of decision outcomes}} \) \times 100

Decision coverage is stronger than statement coverage

100% decision coverage guarantees 100% statement coverage

Not the other way around!
Which of the following statements about the relationship between statement and decision coverage is correct?

a. 100 % decision coverage is achieved if statement coverage is greater than 90 %
b. 100 % statement coverage is achieved if decision coverage is greater than 90 %
c. **100 % decision coverage always means 100 % statement coverage**
d. 100 % statement coverage always means 100 % decision coverage
Question 4

Why are error guessing and exploratory testing techniques good to do?

a. They find defects missed by specification-based and structure-based techniques
b. They don’t require any training to be as effective as formal techniques
c. They can be used more effectively when there are good specifications
d. They will ensure that all of the code or system is tested
Question 4: Clues

Why are error guessing and exploratory testing techniques good to do?

Experience-based techniques
Question 4: Clues

Why are error guessing and exploratory testing techniques good to do?

Error-guessing and Exploratory testing

Experience-based techniques

Error-guessing

Guess: “Where are the defects more likely to be found?”

Anticipate defects based on previous experience

Should always be used as a complement to more formal test techniques

Success depends on skill of the tester → Can be highly effective
Question 4: Clues

Why are error guessing and exploratory testing techniques good to do?

Exploratory testing

- Hands-on approach
- Concurrent test design / execution / logging / learning
- Testers involved in minimum planning and maximum test execution
- Approach is useful when
  - Specification is poor / or does not exist at all
  - Time is limited
- Can complement more formal testing → Ensure most serious defects are found
Why are error guessing and exploratory testing techniques good to do?

a. They find defects missed by specification-based and structure-based techniques
b. They don’t require any training to be as effective as formal techniques
c. They can be used more effectively when there are good specifications
d. They will ensure that all of the code or system is tested
Question 5

How do experience-based techniques differ from specification-based techniques?

a. They depend on the tester’s understanding of the way the system is structured rather than on a documented record of what the system should do
b. They depend on having older testers rather than younger testers
c. They depend on a documented record of what the system should do rather than on an individual’s personal view
d. They depend on an individual’s personal view rather than on a documented record of what the system should do
How do experience-based techniques differ from specification-based techniques?

Experience-based techniques

Tests derived from skill / knowledge / experience / intuition

- Both of technical and business people
- Different groups yield different perspectives

Often based on similar applications and technologies

Used predominantly to complement more formal test techniques

Specification-based and structure-based techniques

Success / Effectiveness is highly dependent on the testers skill and experience
How do experience-based techniques differ from specification-based techniques?

a. They depend on the tester’s understanding of the way the system is structured rather than on a documented record of what the system should do
b. They depend on having older testers rather than younger testers
c. They depend on a documented record of what the system should do rather than on an individual’s personal view
d. **They depend on an individual’s personal view rather than on a documented record of what the system should do**
**Question 6**

**Pair the following test design techniques with the typical problems they address:**

<table>
<thead>
<tr>
<th>Technique</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Decision tables</td>
<td>Applied when the inputs or outputs can be grouped in a way that exhibits similar behaviour</td>
</tr>
<tr>
<td>Use case testing</td>
<td>Used to test sequences of states or sequences of transitions</td>
</tr>
<tr>
<td>State transition testing</td>
<td>Used when the problem can be described as an interaction between an actor and the system</td>
</tr>
<tr>
<td>Boundary value analysis</td>
<td>Used when the inputs and actions can be expressed as Boolean values</td>
</tr>
<tr>
<td>Equivalence partitioning</td>
<td>Applied when the inputs and outputs can be grouped in equivalent partitions. The technique tests the edges of each equivalence partition</td>
</tr>
</tbody>
</table>
Pair the following **test design techniques** with the typical problems they address:

<table>
<thead>
<tr>
<th>Test Design Technique</th>
<th>Typical Problems</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Decision tables</strong></td>
<td>Applied when the <strong>inputs</strong> or <strong>outputs</strong> can be grouped in a way that exhibits similar behaviour</td>
</tr>
<tr>
<td><strong>Use case testing</strong></td>
<td>Used to test <strong>sequences</strong> of <strong>states</strong> or sequences of <strong>transitions</strong></td>
</tr>
<tr>
<td><strong>State transition testing</strong></td>
<td>Used when the <strong>problem</strong> can be described as an <strong>interaction</strong> between an <strong>actor</strong> and the <strong>system</strong></td>
</tr>
<tr>
<td><strong>Boundary value analysis</strong></td>
<td>Used when the <strong>inputs</strong> and <strong>actions</strong> can be expressed as <strong>Boolean values</strong></td>
</tr>
<tr>
<td><strong>Equivalence partitioning</strong></td>
<td>Applied when the <strong>inputs</strong> and <strong>outputs</strong> can be grouped in <strong>equivalent</strong> partitions. The technique <strong>tests</strong> the <strong>edges</strong> of each equivalence partition</td>
</tr>
</tbody>
</table>
If you are flying with an economy ticket, there is a possibility that you may get upgraded to business class, especially if you hold a gold card in the airline’s frequent flyer program.

If you don’t hold a gold card, there is a possibility that you will get “bumped” off the flight if it is full and you check in late.

This is shown in the following figure. Note that each box (i.e. statement) has been numbered.
Question 7

Tests run:

Test 1
Gold card holder who gets upgraded to business class

Test 2
Non-gold card holder who stays in economy

Test 3
A person who is bumped off the flight
Question 7

What is the **statement coverage** of these three **tests**?

a. 60 %
b. 70 %
c. 80 %
d. 90 %
Question 7: Clues

What is the **statement coverage** of these three tests?

Calculating **statement coverage**

\[
\text{Statement coverage} = \frac{\text{Number of statements exercised}}{\text{Total number of statements}} \times 100
\]

**After running all three tests:**

Numerator: How many **statements** have we **exercised**?

Denominator: How many **statements** exist in **total**?

**Multiply by a hundred to get percentage**
Question 7: Clues

What is the statement coverage of these three tests?

Test 1
Gold card holder
Upgraded to business class

Coverage
Total statements: **10**
Statements so far: **4**
Question 7: Clues

What is the **statement coverage** of these three tests?

Test 2
- Non-gold card holder
- Stays in economy

Coverage
- Total statements: 10
- Statements so far: 6
Question 7: Clues

What is the **statement coverage** of these three tests?

**Test 3**
- Any person
- Bumped off the flight

**Coverage**
- Total statements: 10
- Statements so far: 8
Question 7: Clues

What is the **statement coverage** of these three tests?

Calculating statement coverage

How many statements have we exercised? 8


How many statements exist in total? 10

Have yet to exercise statements


Statement coverage = 80 %
Question 7: Answer

What is the *statement coverage* of these three *tests*?

- a. 60 %
- b. 70 %
- c. 80 %
- d. 90 %
Question 8

When **choosing which technique to use** in a given situation, which **factors** should be taken into **account**?

1. Previous experience of types of defects found in this or similar system
2. The existing knowledge of the testers
3. Regulatory standards that apply
4. The type of test executing tool that will be used
5. The documentation available
6. Previous experience in the development language

a. 2, 3, 5, and 6
b. 1, 2, 3 and 5
c. 1, 4 and 5
d. 2, 3 and 5
Question 8: Clues

When choosing which technique to use in a given situation, which factors should be taken into account?

Which technique is best? → Wrong question

Each technique is good for certain instances, and less adequate for others

“The best testing technique is no single testing technique”

Examples

Structure-based → Can only test what is present
   E.g. find malicious code / Trojan horses

Specification-based → Can reveal if parts of specification are missing from code

Experience-based → Finds things missing from both specification and code
Question 8: Clues

When choosing which technique to use in a given situation, which factors should be taken into account?

Each technique is aimed at particular types of defects

E.g. State-transition testing is unlikely to find boundary defects

Use a variety of testing techniques

Using one technique → Ensures many defects of that particular class are found

However → Ensures many defects of other classes are missed

Using a variety of techniques

Ensures a variety of defects are found

Effective testing
Question 8: Clues

When **choosing which technique to use** in a given situation, which **factors** should be taken into **account**?

**Internal factors** affecting choice of test techniques

- **Models used**
  - If specification contains state transition diagram → State transition testing
- **Testers knowledge and experience**
  - How much do testers know about the system / various techniques?
- **Likely defects**
  - Each technique is good at finding particular defects
  - Knowledge about likely defects is therefore helpful
Question 8: Clues

When **choosing which technique to use** in a given situation, which **factors** should be taken into **account**?

**Internal factors** affecting choice of test techniques

**Test objective**
- What do we **want** from the test **effort**? → Helps us define **approach**

**Documentation**
- Exists? Updated? Content → Serves to **guide** the test **effort**

**Life cycle model**
- Sequential → **Formal** testing techniques
- Iterative → **Exploratory** testing approach
**Question 8: Clues**

When **choosing which technique to use** in a given situation, which **factors** should be taken into **account**?

**External factors affecting choice of test techniques**

- **Risk**
  
  The **greater** the risk, the **greater** the need for more **thorough** testing

- **Customer / Contractual requirements**
  
  Contracts may **specify** particular testing **techniques** to be used

- **Type of system**
  
  **Influence** techniques used
  
  **E.g.** Financial application → Benefits from boundary value analysis
Question 8: Clues

When **choosing which technique to use** in a given situation, which **factors** should be taken into **account**?

**External factors affecting choice of test techniques**

- **Regulatory requirements**
  - Some industries have regulatory standards
  - **E.g.** Aircraft industry → Test effort depends on level of SW integrity required
    - Equivalence partitioning / BVA / State transition
    - Combined with statement / decision coverage

- **Time and budget**
  - How much time is available? More time → More techniques
Question 8: Answer

When choosing which technique to use in a given situation, which factors should be taken into account?

1. Previous experience of types of defects found in this or similar system
2. The existing knowledge of the testers
3. Regulatory standards that apply
4. The type of test executing tool that will be used
5. The documentation available
6. Previous experience in the development language

a. 2, 3, 5, and 6
b. 1, 2, 3 and 5
c. 1, 4 and 5
d. 2, 3 and 5
Question 9

Given the state diagram below, which test case is the minimum series of valid transitions to cover every state?

a. SS-S1-S2-S4-S1-S3-ES
b. SS-S1-S2-S3-S4-ES
c. SS-S1-S2-S4-S1-S3-S4-S1-S3-ES
d. SS-S1-S4-S2-S1-S3-ES
Question 9: Clues

Given the state **diagram** below, which **test case** is the **minimum series of valid transitions** to cover every state?

Want the minimum path from SS to ES, visiting each state at least once

Path:
Question 9: Clues

Given the state diagram below, which test case is the minimum series of valid transitions to cover every state?

Want the minimum path from SS to ES, visiting each state at least once

Path:
SS-
Question 9: Clues

Given the state **diagram** below, which **test case** is the minimum series of valid transitions to cover every state?

Want the minimum path from SS to ES, visiting each state at least once

Path: SS-S1
Question 9: Clues

Given the state diagram below, which test case is the minimum series of valid transitions to cover every state?

Want the minimum path from SS to ES, visiting each state at least once

Path: SS-S1-S2
Given the state **diagram** below, which **test case** is the **minimum** series of **valid transitions** to cover every state?

Want the minimum path from SS to ES, visiting each state at least once

Path:
SS-S1-S2-S4
Given the state diagram below, which test case is the minimum series of valid transitions to cover every state?

Want the minimum path from SS to ES, visiting each state at least once.

Path: SS-S1-S2-S4-S1
Question 9: Clues

Given the state diagram below, which test case is the minimum series of valid transitions to cover every state?

Want the minimum path from SS to ES, visiting each state at least once

Path:
SS-S1-S2-S4-S1-S3
Question 9: Clues

Given the state **diagram** below, which **test case** is the **minimum** series of **valid transitions** to cover every **state**?

Want the minimum path from SS to ES, visiting each state at least once

Path:
SS-S1-S2-S4-S1-S3-ES
Given the state diagram below, which test case is the minimum series of valid transitions to cover every state?

a. SS-S1-S2-S4-S1-S3-ES
b. SS-S1-S2-S3-S4-ES
c. SS-S1-S2-S4-S1-S3-S4-S1-S3-ES
d. SS-S1-S4-S2-S1-S3-ES
Part II: Exercises and Open-ended questions
Exercise 1: Decision Table Testing

If you hold an “over 60s” rail card, you get a 34% discount on whatever ticket you buy. If you are travelling with a child (under 16) you get a 50% discount on any ticket if you hold a family rail card, otherwise you get a 10% discount. You may only hold one type of rail card.

a. Produce a decision table showing all combinations of fare types and resulting discounts
b. Derive test cases from the decision table
Exercise 1(a): Clues

Produce a decision table showing all combinations of fare types and resulting discounts

Fare types available based on:

“Over 60s” card
Family card
Travelling with a child

Set up the decision table

Three different conditions → Card type
Each with the outcome Y / N → Holds said card type / Does not hold said card type
Eight different rules → Maps out combinations between inputs and outputs
Exercise 1(a): Clues

Produce a decision table showing all combinations of fare types and resulting discounts

**Decision table for fare types and discounts**

Each rule yields a different effect

<table>
<thead>
<tr>
<th>Causes (Inputs)</th>
<th>R1</th>
<th>R2</th>
<th>R3</th>
<th>R4</th>
<th>R5</th>
<th>R6</th>
<th>R7</th>
<th>R8</th>
</tr>
</thead>
<tbody>
<tr>
<td>Over 60s rail card?</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>Family rail card?</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>Child also travelling?</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td>N</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Effects (Outputs)</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Discount (%)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Exercise 1(a): Clues

Produce a decision table showing all combinations of fare types and resulting discounts

**Decision table**

R8: No rail cards and no children travelling

<table>
<thead>
<tr>
<th>Causes (Inputs)</th>
<th>R1</th>
<th>R2</th>
<th>R3</th>
<th>R4</th>
<th>R5</th>
<th>R6</th>
<th>R7</th>
<th>R8</th>
</tr>
</thead>
<tbody>
<tr>
<td>Over 60s rail card?</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>Family rail card?</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>Child also travelling?</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td>N</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Effects (Outputs)</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Discount (%)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0%</td>
</tr>
</tbody>
</table>


Exercise 1(a): Clues

Produce a decision table showing all combinations of fare types and resulting discounts

Decision table

R7: No rail cards, but travelling with children

<table>
<thead>
<tr>
<th>Causes (Inputs)</th>
<th>R1</th>
<th>R2</th>
<th>R3</th>
<th>R4</th>
<th>R5</th>
<th>R6</th>
<th>R7</th>
<th>R8</th>
</tr>
</thead>
<tbody>
<tr>
<td>Over 60s rail card?</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>Family rail card?</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>Child also travelling?</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td>N</td>
</tr>
<tr>
<td>Effects (Outputs)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Discount (%)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td><strong>10%</strong></td>
</tr>
</tbody>
</table>
Exercise 1(a): Clues

Produce a **decision table** showing all combinations of fare types and resulting **discounts**

**Decision table**

R6: No “Over 60s” card, with family card, but no children travelling

<table>
<thead>
<tr>
<th>Causes (Inputs)</th>
<th>R1</th>
<th>R2</th>
<th>R3</th>
<th>R4</th>
<th>R5</th>
<th>R6</th>
<th>R7</th>
<th>R8</th>
</tr>
</thead>
<tbody>
<tr>
<td>Over 60s rail card?</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>Family rail card?</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>Child also travelling?</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td>N</td>
</tr>
<tr>
<td>Effects (Outputs)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Discount (%)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0%</td>
<td>10%</td>
<td>0%</td>
</tr>
</tbody>
</table>
Exercise 1(a): Clues

Produce a **decision table** showing all combinations of fare types and resulting **discounts**

**Decision table**

R5: No “Over 60s” card, with family card, and with children travelling

<table>
<thead>
<tr>
<th>Causes (Inputs)</th>
<th>R1</th>
<th>R2</th>
<th>R3</th>
<th>R4</th>
<th>R5</th>
<th>R6</th>
<th>R7</th>
<th>R8</th>
</tr>
</thead>
<tbody>
<tr>
<td>Over 60s rail card?</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>Family rail card?</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>Child also travelling?</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td>N</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Effects (Outputs)</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Discount (%)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>50%</td>
<td>0%</td>
<td>10%</td>
<td>0%</td>
</tr>
</tbody>
</table>
Exercise 1(a): Clues

Produce a decision table showing all combinations of fare types and resulting discounts

**Decision table**

R4: Holds “Over 60s” card, no family card, and no children travelling

<table>
<thead>
<tr>
<th>Causes (Inputs)</th>
<th>R1</th>
<th>R2</th>
<th>R3</th>
<th>R4</th>
<th>R5</th>
<th>R6</th>
<th>R7</th>
<th>R8</th>
</tr>
</thead>
<tbody>
<tr>
<td>Over 60s rail card?</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>Family rail card?</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>Child also travelling?</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td>N</td>
</tr>
</tbody>
</table>

**Effects (Outputs)**

| Discount (%) | 34% | 50% | 0%  | 10% | 0%  |
Exercise 1(a): Clues

Produce a **decision table** showing all **combinations** of fare types and resulting **discounts**

**Decision table**

R3: Holds “Over 60s” card, no family card, but with children travelling

<table>
<thead>
<tr>
<th>Causes (Inputs)</th>
<th>R1</th>
<th>R2</th>
<th>R3</th>
<th>R4</th>
<th>R5</th>
<th>R6</th>
<th>R7</th>
<th>R8</th>
</tr>
</thead>
<tbody>
<tr>
<td>Over 60s rail card?</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>Family rail card?</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>Child also travelling?</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td>N</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Effects (Outputs)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Discount (%)</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>34%</td>
</tr>
<tr>
<td>34%</td>
</tr>
<tr>
<td>50%</td>
</tr>
<tr>
<td>0%</td>
</tr>
<tr>
<td>10%</td>
</tr>
<tr>
<td>0%</td>
</tr>
</tbody>
</table>
Exercise 1(a): Clues

Produce a decision table showing all combinations of fare types and resulting discounts

<table>
<thead>
<tr>
<th>Causes (Inputs)</th>
<th>R1</th>
<th>R2</th>
<th>R3</th>
<th>R4</th>
<th>R5</th>
<th>R6</th>
<th>R7</th>
<th>R8</th>
</tr>
</thead>
<tbody>
<tr>
<td>Over 60s rail card?</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>Family rail card?</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>Child also travelling?</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td>N</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Effects (Outputs)</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Discount (%)</td>
<td>X/?/34%</td>
<td>34%</td>
<td>34%</td>
<td>50%</td>
<td>0%</td>
<td>10%</td>
<td>0%</td>
<td></td>
</tr>
</tbody>
</table>

R2: Holds “Over 60s” card, has family card, but no children travelling
Exercise 1(a): Clues

Produce a decision table showing all combinations of fare types and resulting discounts

**Decision table**

R1: Holds “Over 60s” card, has family card, with children travelling

<table>
<thead>
<tr>
<th>Causes (Inputs)</th>
<th>R1</th>
<th>R2</th>
<th>R3</th>
<th>R4</th>
<th>R5</th>
<th>R6</th>
<th>R7</th>
<th>R8</th>
</tr>
</thead>
<tbody>
<tr>
<td>Over 60s rail card?</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>Family rail card?</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>Child also travelling?</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td>N</td>
</tr>
<tr>
<td>Effects (Outputs)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Discount (%)</td>
<td>X/?/50%</td>
<td>X/?/34%</td>
<td>34%</td>
<td>34%</td>
<td>50%</td>
<td>0%</td>
<td>10%</td>
<td>0%</td>
</tr>
</tbody>
</table>
Exercise 1(a): Clues

Produce a decision table showing all combinations of fare types and resulting discounts

**Decision table**

Final result → What is the output for R1 and R2?

<table>
<thead>
<tr>
<th>Causes (Inputs)</th>
<th>R1</th>
<th>R2</th>
<th>R3</th>
<th>R4</th>
<th>R5</th>
<th>R6</th>
<th>R7</th>
<th>R8</th>
</tr>
</thead>
<tbody>
<tr>
<td>Over 60s rail card?</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>Family rail card?</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>Child also travelling?</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td>N</td>
</tr>
<tr>
<td>Effects (Outputs)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Discount (%)</td>
<td>X/50%</td>
<td>X/34%</td>
<td>34%</td>
<td>34%</td>
<td>50%</td>
<td>0%</td>
<td>10%</td>
<td>0%</td>
</tr>
</tbody>
</table>
Exercise 1(a): Clues

Produce a **decision table** showing all **combinations of fare types** and resulting **discounts**

What is the **output** for R1 and R2?

- X → Not possible to hold more than one rail card
- ? → Specification does not tell us what happens for said cases

If someone holds **two cards** → Not likely to admit that

- **R1**: Claim **50% discount** with family rail card and travelling with children
- **R2**: Claim **34 % discount** with “Over 60s” card and no children

Notation shows we **do not know** the expected **outcome** for R1 and R2

Have revealed **ambiguities** in the **specification**
Exercise 1(a): Clues

Produce a decision table showing all combinations of fare types and resulting discounts

Further simplifications

R3 and R4 → Same effect (34% discount)
  Third cause (children also travelling?) has no effect on the outcome

R6 and R8 → Same effect (0% discount)
  Having family rail card has no effect when not travelling with a child

Rationalise table

Combine these with a “not applicable” entry

Fewer columns and fewer test cases
Exercise 1(a): Answer

Produce a decision table showing all combinations of fare types and resulting discounts

Rationalised decision table

R3*: Combined rules 3 and 4 / R6*: Combined rules 6 and 8

<table>
<thead>
<tr>
<th>Causes (Inputs)</th>
<th>R1</th>
<th>R2</th>
<th>R3*</th>
<th>R5</th>
<th>R6*</th>
<th>R7</th>
</tr>
</thead>
<tbody>
<tr>
<td>Over 60s rail card?</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>Family rail card?</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td>-</td>
<td>N</td>
</tr>
<tr>
<td>Child also travelling?</td>
<td>Y</td>
<td>N</td>
<td>-</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Effects (Outputs)</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Discount (%)</td>
<td>50%</td>
<td>34%</td>
<td>34%</td>
<td>50%</td>
<td>0%</td>
<td>10%</td>
</tr>
</tbody>
</table>
Derive **test cases from the decision table**

**Test cases for rail card scenario**

<table>
<thead>
<tr>
<th>Test case ID</th>
<th>Input</th>
<th>Expected outcome</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>A. Adams, with over 60s rail card and family rail card, travelling with grandson Ben (age 11).</td>
<td>50% discount for both tickets</td>
</tr>
<tr>
<td>2</td>
<td>Mrs. B. Cook, with over 60s rail card and family rail card, travelling alone.</td>
<td>34% discount</td>
</tr>
<tr>
<td>3</td>
<td>Mr. J. Johnson, with over 60s rail card, travelling with his wife.</td>
<td>34% discount (for Johnson only, not the wife)</td>
</tr>
<tr>
<td>4</td>
<td>Mrs. C. Baker, with family rail card, travelling with her daughter Anna.</td>
<td>50% discount for both tickets</td>
</tr>
<tr>
<td>5</td>
<td>Miss A. Lone, no rail card, travelling alone</td>
<td>No discount</td>
</tr>
<tr>
<td>6</td>
<td>Mr. J. Harper, with no rail card, travelling with his niece (age 5)</td>
<td>10% discount for both tickets</td>
</tr>
</tbody>
</table>
Exercise 1(b): Answer

Derive **test cases** from the **decision table**

**Additional issues?**

Does **discount** apply only to the **traveller**, or to **someone** travelling **with** them?

**Specification** does not explicitly state the answer

**Assumptions made**

**Family card**: Discounts apply to **all travelling members**

**Over 60s card**: Discount **only** applies to the **individual** passenger
Exercise 2: State Transitions

A website **shopping basket** starts out **empty**. As **purchases** are **selected**, they are **added** to the shopping basket. **Items** can also be **removed** from the shopping basket.

When the customer **decides to check out**, a **summary** of the items in the basket and the **total cost** are **show**. Customer states if the information is OK.

*If the contents and the price are OK, then you leave the summary display and go to the payment system. Otherwise, you go back to shopping (so as to remove items if you want).*

a.  (i)Produce a **state diagram** showing the different **states** and **transitions**.
   (ii)Define a **test**, in terms of a **sequence** of **states**, to cover all **transitions**

b. Produce a **state table**. Give an **example test** for an **invalid transition**
Exercise 2(a.i): State Transitions

Produce a state diagram showing different states and transitions
Exercise 2(a.i): Clues

Produce a state diagram showing different states and transitions

1. Mapping out the different states

   - **Empty** (start state)
     - Nothing has been placed into the basket

   - **Shopping** (intermediate)
     - There are items placed in the basket

   - **Summary and cost** (intermediate)
     - Overview of the items and price

   - **Payment** (final state)
Exercise 2(a.i): Clues

Produce a state diagram showing different states and transitions

2. Mapping out the transitions between states

Add item → “Empty” to “Shopping” or “Shopping” to “Shopping”
Remove item → “Shopping” to “Shopping”
Remove last item → “Shopping” to “Empty”
Check out → “Shopping” to “Summary and cost”
OK → “Summary and cost” to “Payment”
Not OK → “Summary and cost” to “Shopping”
Exercise 2(a.i): Clues

Produce a state diagram showing different states and transitions

3. Model state diagram

Start with the initial state “S1: Empty”

What can you do in this state?
   Add item

Which state do you reach?
   “S2: Shopping”
Exercise 2(a.i): Clues

Produce a **state diagram** showing different **states** and **transitions**

3. **Model state diagram**
   - For the state “S2: Shopping”
     - What can you **do within** this state?
       - Add item / Remove item
     - Which state do you **reach**?
       - Still in “S2: Shopping”
Exercise 2(a.i): Clues

Produce a state diagram showing different states and transitions

3. Model state diagram

State “S2: Shopping”

Interaction with other states?
Remove last item / Check out
Which states do you reach?
“S1: Empty” and “S3: Summary and cost”
Exercise 2(a.i): Clues

Produce a **state diagram** showing different **states** and **transitions**

3. **Model** state diagram

State “S3: Summary and cost”

What can you **do** in this state?

OK / Not OK

Which states do you **reach**?

“S2: Shopping” and “S4: Payment”
Exercise 2(a.i): Answer

Produce a state diagram showing different states and transitions

3. Final diagram

![State Diagram](image-url)
Exercise 2(a.ii): State Transitions

Define a test, in terms of a sequence of states, to cover all transitions
Exercise 2(a.ii): Clues

Define a **test**, in terms of a **sequence of states**, to cover **all transitions**

Find a **path** from **S1** to **S4**, covering **all transitions**

That is pass **every arrow** in the state diagram
Exercise 2(a.ii): Clues

Define a test, in terms of a sequence of states, to cover all transitions.
Exercise 2(a.ii): Clues

Define a test, in terms of a sequence of states, to cover all transitions.
Exercise 2(a.ii): Clues

Define a test, in terms of a sequence of states, to cover all transitions
Define a **test**, in terms of a **sequence of states**, to cover all transitions.
Define a **test**, in terms of a **sequence of states**, to cover all transitions.
Define a **test**, in terms of a **sequence of states**, to cover all transitions.
Define a test, in terms of a sequence of states, to cover all transitions
Define a **test**, in terms of a **sequence of states**, to cover **all transitions**.
Define a test, in terms of a sequence of states, to cover all transitions.
Define a test, in terms of a sequence of states, to cover all transitions
Exercise 2(a.ii): Answer

Define a **test**, in terms of a **sequence of states**, to cover **all transitions**

<table>
<thead>
<tr>
<th>State</th>
<th>Event (action)</th>
</tr>
</thead>
<tbody>
<tr>
<td>S1</td>
<td>Add item</td>
</tr>
<tr>
<td>S2</td>
<td>Remove (last) item</td>
</tr>
<tr>
<td>S1</td>
<td>Add item</td>
</tr>
<tr>
<td>S2</td>
<td>Add item</td>
</tr>
<tr>
<td>S2</td>
<td>Remove item</td>
</tr>
<tr>
<td>S2</td>
<td>Check out</td>
</tr>
<tr>
<td>S3</td>
<td>Not OK</td>
</tr>
<tr>
<td>S2</td>
<td>Check out</td>
</tr>
<tr>
<td>S3</td>
<td>OK</td>
</tr>
<tr>
<td>S4</td>
<td>-</td>
</tr>
</tbody>
</table>
Exercise 2(b): State Transitions

Produce a state table. Give an example test for an invalid transition.
Exercise 2(b): Clues

Produce a **state table**. Give an **example** test for an **invalid transition**

State table

Maps out states and transitions in tabular form

<table>
<thead>
<tr>
<th>State (Event)</th>
<th>Add item</th>
<th>Remove item</th>
<th>Remove last item</th>
<th>Check out</th>
<th>OK</th>
<th>Not OK</th>
</tr>
</thead>
<tbody>
<tr>
<td>S1: Empty</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>S2: Shopping</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>S3: Summary</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>S4: Payment</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Exercise 2(b): Clues

Produce a state table. Give an example test for an invalid transition.

S1: Empty

What states can we reach from S1?

Through which transitions?

<table>
<thead>
<tr>
<th>State (Event)</th>
<th>Add item</th>
<th>Remove item</th>
<th>Remove last item</th>
<th>Check out</th>
<th>OK</th>
<th>Not OK</th>
</tr>
</thead>
<tbody>
<tr>
<td>S1: Empty</td>
<td>S2</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>OK</td>
<td>-</td>
</tr>
<tr>
<td>S2: Shopping</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>S3: Summary</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>S4: Payment</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>
Exercise 2(b): Clues

Produce a state table. Give an example test for an invalid transition

S2: Shopping

What states can we reach from S2?

Through which transitions?

<table>
<thead>
<tr>
<th>State (Event)</th>
<th>Add item</th>
<th>Remove item</th>
<th>Remove last item</th>
<th>Check out</th>
<th>OK</th>
<th>Not OK</th>
</tr>
</thead>
<tbody>
<tr>
<td>S1: Empty</td>
<td>S2</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>S2: Shopping</td>
<td>S2</td>
<td>S2</td>
<td>S1</td>
<td>S3</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>S3: Summary</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>S4: Payment</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Exercise 2(b): Clues

Produce a state table. Give an example test for an invalid transition

S3: Summary and cost

What states can we reach from S3?

Through which transitions?

<table>
<thead>
<tr>
<th>State (Event)</th>
<th>Add item</th>
<th>Remove item</th>
<th>Remove last item</th>
<th>Check out</th>
<th>OK</th>
<th>Not OK</th>
</tr>
</thead>
<tbody>
<tr>
<td>S1: Empty</td>
<td>S2</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>S2: Shopping</td>
<td>S2</td>
<td>S2</td>
<td>S1</td>
<td>S3</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>S3: Summary</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>S2</td>
<td>S4</td>
</tr>
<tr>
<td>S4: Payment</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Exercise 2(b): Answer

Produce a state table. Give an example test for an invalid transition

S4: Payment

What states can we reach from S4?

Through which transitions?

<table>
<thead>
<tr>
<th>State (Event)</th>
<th>Add item</th>
<th>Remove item</th>
<th>Remove last item</th>
<th>Check out</th>
<th>OK</th>
<th>Not OK</th>
</tr>
</thead>
<tbody>
<tr>
<td>S1: Empty</td>
<td>S2</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>S2: Shopping</td>
<td>S2</td>
<td>S2</td>
<td>S1</td>
<td>S3</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>S3: Summary</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>S2</td>
<td>S4</td>
</tr>
<tr>
<td>S4: Payment</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>
Exercise 3: Statement and Decision

A vending machine dispenses either hot or cold drinks.

If you choose a hot drink (e.g. tea or coffee), it asks if you want milk (added if required).

Ten it asks if you want sugar (added if required)

Finally, the drink is dispensed.
Exercise 3(a)

Draw a control flow diagram for this example

Hint: Regard the selection of the type of drink as one statement
Exercise 3(a): Clues

Draw a control flow diagram for this example

1. Map out the different controls for the scenario
   - What is being controlled?
     - Hot or cold drink
     - Milk or no milk
     - Sugar or no sugar

2. Represent each control with a diamond shape
   - These controls will lead to decisions
   - E.g. Choosing a “hot” drink, or choosing “no milk”.
Exercise 3(a): Clues

Draw a control flow diagram for this example

3. Map out the different outcomes (statements) for each control

What are the outcomes?
- Hot drink → Select drink (coffee or tea)
- Cold drink → Select drink (water or soda)
- Milk → Add milk
- No milk → Nothing happens
- Sugar → Add sugar
- No sugar → Nothing

4. Represent the statements with rectangles
Exercise 3(a): Clues

Draw a control flow diagram for this example

5. Map out the different *decisions* (statements) for each control

What are the outcomes of each *question*?

Which statements do they lead to?

6. Represent the outcomes with arrows in the diagram
Exercise 3(a): Answer

Draw a control flow diagram for this example

7. Construct the control flow diagram combining all elements

```
Hot or cold?

Select drink (water or soda)

Milk?

Sugar?

Dispense drink

Select drink (coffee or tea)

Add milk

Add sugar
```

Yes

No

Yes

No

Hot

Cold
Exercise 3(b)

Given the following tests, what is the statement coverage achieved? What is the decision coverage achieved?

Test 1: Cold drink
Test 2: Hot drink with milk and sugar
Exercise 3(b): Clues

Statement and decision coverage

Test 1:
Cold drink

Test 2:
Hot drink with milk and sugar

What is the statement coverage?
What is the decision coverage?
Statement and decision coverage

Statement coverage
100% statement coverage
Every statement has been covered
(All boxes have been touched)

What is the decision coverage?

How many decision outcomes exist?
How many decision outcomes exercised?
Statement and decision coverage

What is the decision coverage?

How many decision outcomes exist?
Hot / Cold / Yes / No / Yes / No
6 decision outcomes in total

How many decision outcomes exercised?
Hot / Cold / Yes / No
4 decision outcomes exercised

Decision coverage
4 / 6 = 67 %
Exercise 3(c)

What **additional tests** would be **needed** to achieve **100% decision and statement coverage**?
Exercise 3(c): Answer

What **additional tests** would be needed to achieve **100% decision and statement coverage**?

**Additional tests**

- **Statement coverage:**
  - No further tests

- **Decision coverage**
  - Must exercise No / No

- **Test 3:**
  - Hot drink, no milk, no sugar
  - *All decisions exercised*
The slides are made by

Yulai Fjeld, revised by Eva H. Vihovde