



CHALMERS
UNIVERSITY OF TECHNOLOGY



UNIVERSITY OF GOTHENBURG

Exercise Session: System Testing

Gregory Gay
DIT635 - February 5, 2021



Finish Activities (Lec 6) First!

Airport Connection Check

<https://bit.ly/3L6vWP9>

- API function to check validity of a connection between two flights.
 - If the arrival airport of Flight A differs from the departure airport of Flight B, connection is invalid.
 - If departure time of Flight B is too close to the arrival time of Flight A, connection is invalid.
 - If an airport doesn't exist, the connection is invalid...



Airport Connection Check

<https://bit.ly/3L6vWP9>

```
validConnection(Flight flightA, Flight flightB)  
returns ValidityCode
```

A **Flight** is a data structure consisting of:

- A unique identifying flight code (string, three characters followed by four numbers).
- The originating airport code (three character string).
- The scheduled departure time from the originating airport (in universal time).
- The destination airport code (three character string).
- The scheduled arrival time at the destination airport (in universal time).

Airport Connection Check

<https://bit.ly/3L6vWP9>

There is also a flight database, where each record contains:

- Three-letter airport code (three character string).
- Airport country (string).
 - If in the Schengen Area, this is indicated instead of the home country.
- Minimum domestic connection time
 - (integer, minimum num. minutes that must be allowed for flight connections to be valid).
- Minimum international connection time
 - (more time is required due to need to clear customs and meet regulations)

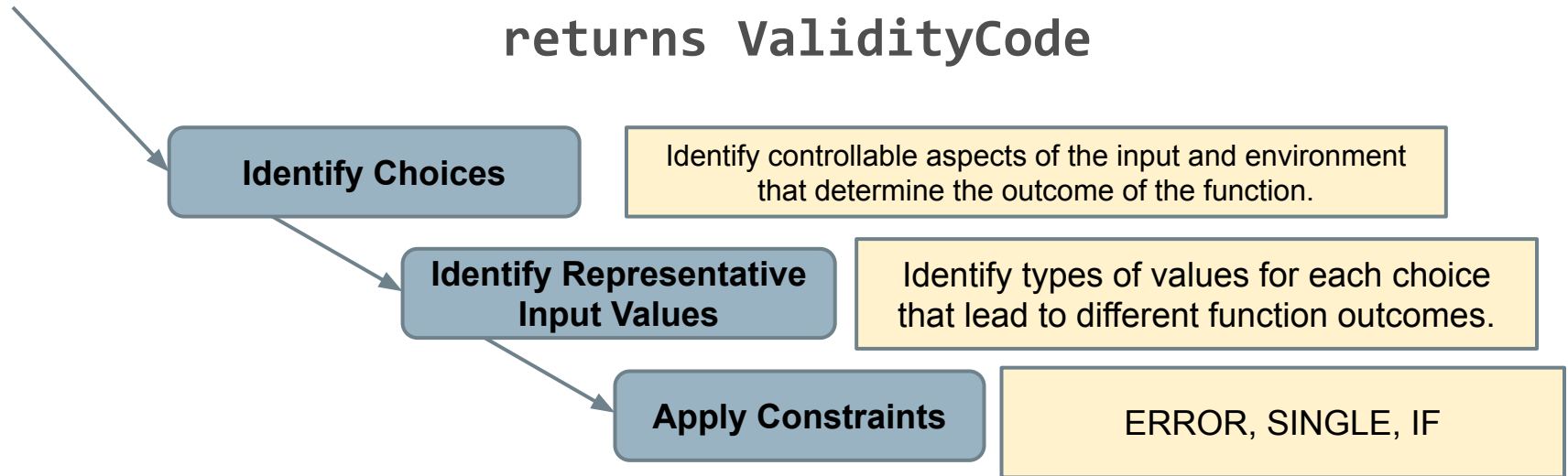
ValidityCode is an integer with value:

- 0 for OK
- 1 for invalid airport code
- 2 for a connection that is too short
- 3 for flights that do not connect (flightA does not land in same location as flightB)
- 4 for any other errors (malformed input or any other unexpected errors).

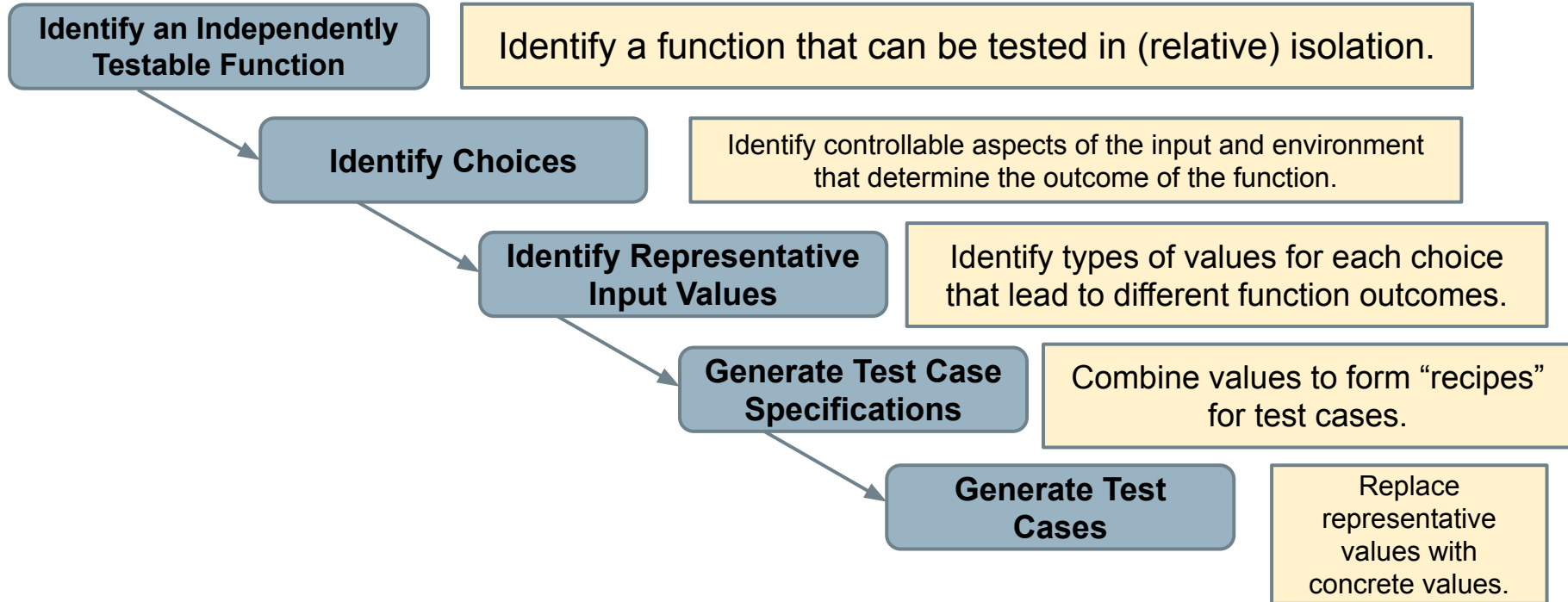
Your Task

<https://bit.ly/3L6vWP9>

`validConnection(Flight flightA, Flight flightB)`
returns `ValidityCode`



Creating System-Level Tests



Hints

<https://bit.ly/3L6vWP9>

- Two explicit parameters (Flight A and B) and one implicit (airport database).
 - Flight has multiple fields (potential choices)
 - Database records have multiple fields (potential choices).
 - Remember that representative values can interact. This must be accounted for.
 - **IF constraints indicate when combinations of values should be used for different choices.**

Hints

<https://bit.ly/3L6vWP9>

- Consider how arrival time (flight A), departure time (flight B), and minimum connection time interact.
- Consider that domestic and international connection times can differ in length.
- Consider how the database contents can influence behavior.
- Consider how input can be invalid or malformed
 - (don't just list "invalid input" but give clear examples).

Example to Start

<https://bit.ly/3L6vWP9>

FlightA

Choice: Originating Airport Code

- Valid airport
- Not in database **[error]**
- Not a correctly formatted airport **[error]**
 - (not a three-letter string)

FlightB

Choice: Originating Airport Code

- Valid airport, same as FlightA's Destination Airport Code
- Valid airport, but different from FlightA's Destination Airport Code **[error]**
- Not in database **[error]**
- Not a correctly formatted airport **[error]**

Current Status

Activity:

<https://bit.ly/3L6vWP9>

I and the TAs are available to answer questions.

- Afonso Fontes
- Sandra Eisenberg
- Chaneli Silva



UNIVERSITY OF
GOTHENBURG



CHALMERS
UNIVERSITY OF TECHNOLOGY