

# **Tutorial 3**

**Swinburne University of Technology**

Software Testing and Reliability (SWE30009)

Semester 2, 2023

Lecturer: Prof T. Y. Chen

Tutor: Dr Hung Q Luu



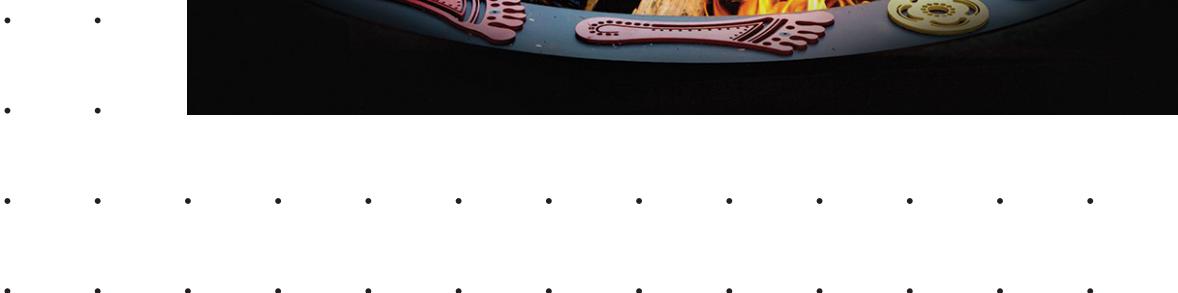
## Acknowledgement of Country

We respectfully acknowledge the Wurundjeri People of the Kulin Nation, who are the Traditional Owners of the land on which Swinburne's Australian campuses are located in Melbourne's east and outer-east, and pay our respect to their Elders past, present and emerging.

We are honoured to recognise our connection to Wurundjeri Country, history, culture, and spirituality through these locations, and strive to ensure that we operate in a manner that respects and honours the Elders and Ancestors of these lands.

We also respectfully acknowledge Swinburne's Aboriginal and Torres Strait Islander staff, students, alumni, partners and visitors.

We also acknowledge and respect the Traditional Owners of lands across Australia, their Elders, Ancestors, cultures, and heritage, and recognise the continuing sovereignties of all Aboriginal and Torres Strait Islander Nations.



Part A

Tutorial task

# Metamorphic relation

- Consider a program which returns the **minimum for a series of integers**.
- Define **four MRs** for this program using the notation and description template discussed in the lecture notes.

# Part B

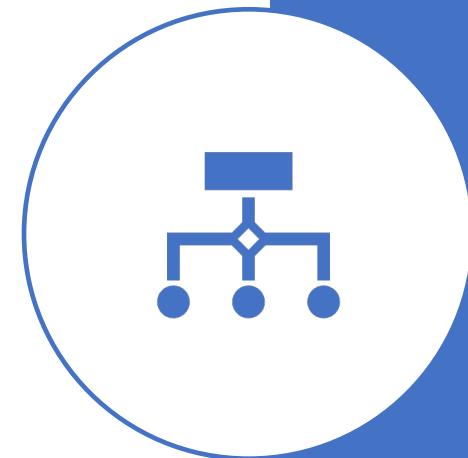
## Revisiting the Lectures

## Testing versus faults

Testing can demonstrate the presence of faults but **not the absence of faults**

# (Un)testable program

- A program is said to be **testable** if the output of any input can be verified
- A program is said to be **untestable** or nontestable if the outputs of some inputs cannot be verified



# Test oracle

Test oracle is a mechanism or procedure against which the computed **outputs could be verified**

## Test oracle problem

- Absence of test oracle
- Test oracles are available but too expensive to be applied



# Traditional testing

- Can only work if test oracle does exist
- Test oracle is used to verify individual test cases

# Metamorphic testing

- Can work even if test oracle does not exist
- Makes use of an identified relation among **multiple test cases**
  - This relationship is referred to as Metamorphic Relation (MR)

## Discussion

- Propose your own MR to test program

$$\mathbf{y = \sin(x)}$$

## Discussion

- Propose your own MR to test program

$$y = 1 + 2 + \dots + n$$

## Discussion

- Propose your own MR to test program

$$y = 1^2 + 2^2 + \dots + n^2$$

# Testing process

1

Identify a  
**metamorphic  
relation (MR)**

2

Define and execute  
**source test cases**

3

Construct and  
execute **follow-up  
test cases** using MR

4

**Verify MR** using  
both source and  
follow-up outputs  
(and if needed, the  
inputs)

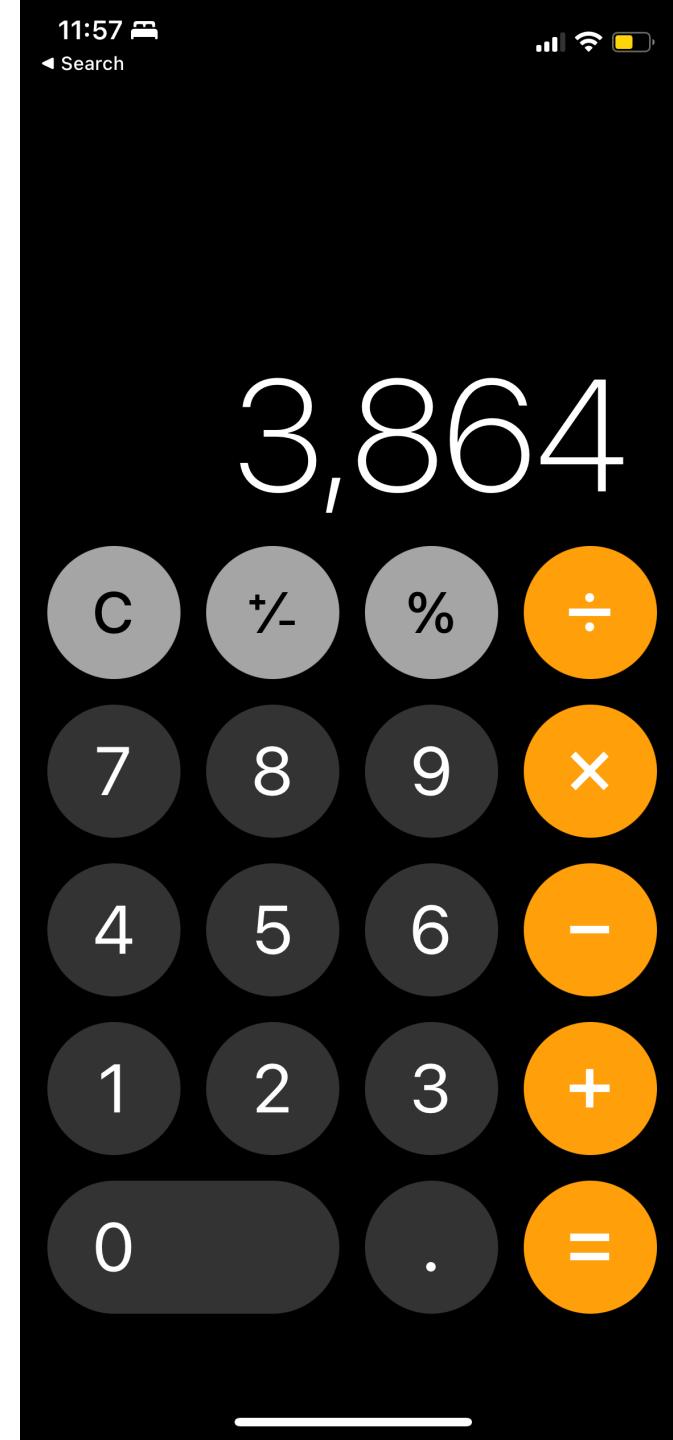
## Discussion $y = \sin(x)$

1. MR:  $\sin(x) = \sin(360^\circ + x)$
2. Source test case
  - Input:  $15^\circ$
  - Output:  $\sin(15^\circ) \sim 0.25$
3. Follow-up test case
  - Input:  $15^\circ + 360^\circ = 375^\circ$
  - Output:  $\sin(375^\circ) \sim 0.17$
4. Verify MR:  $0.25 \stackrel{?}{=} 0.17$

# Calculator App

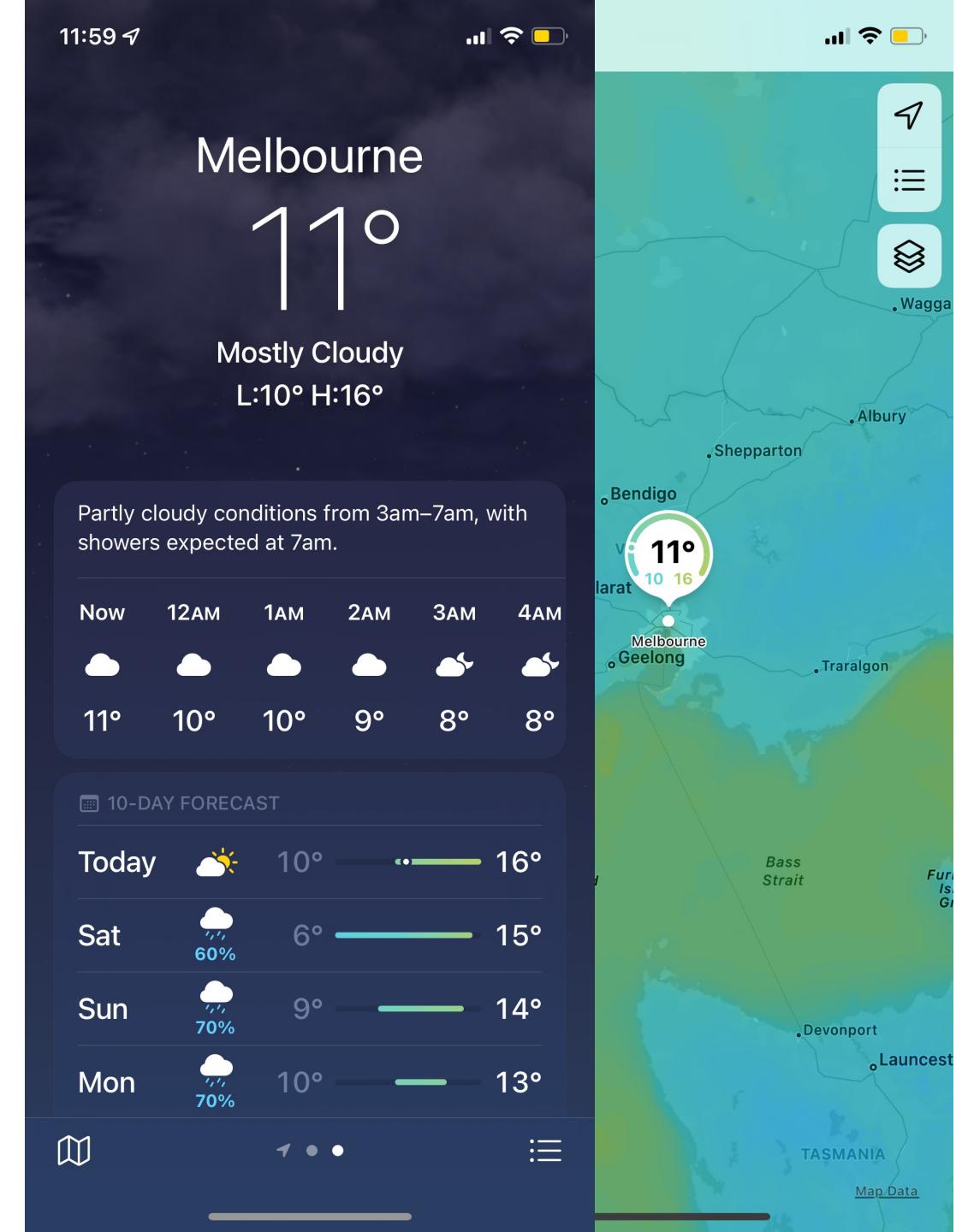
- Which method is more appropriate?
  - Conventional testing
  - Metamorphic testing

**Assumption:** The calculator is to do 1 operation only.



# Weather App

- Which method is more appropriate?
  - Conventional testing
  - Metamorphic testing



# Vending machine

- Which method is more appropriate?
  - Conventional testing
  - Metamorphic testing

