# INUNIVERSITY OF CANBERRA

## INTRODUCTION TO INFORMATION TECHNOLOGY (4478/8936)

### FINAL ASSIGNMENT – 4478/8936

## Instructions

This final assignment will test your knowledge and skills in writing a robust application in Python and producing a minimum set of tests to assess a piece of code according to certain business rules. As such, the assignment requires you to integrate and synthesise what you have learnt so far in this unit, in order to design and create a proper working solution, and to perform a basic software testing task.

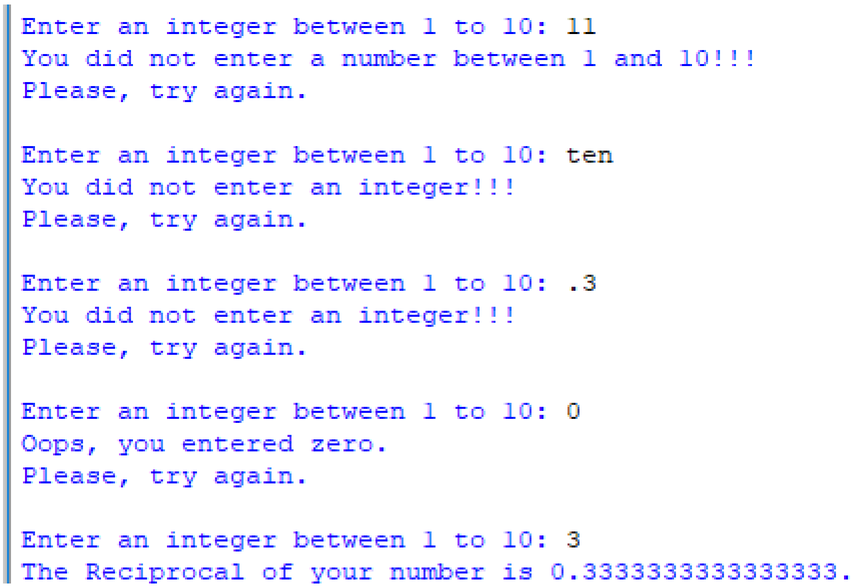
This assessment contributes towards 20% of the total marks allocated in the unit. It has two questions: one on Exception Handling and one on Software Testing. Marks are evenly distributed. That is, 50% question one and 50% question two.

## Submission

Question 1 requires you to complete a Python code written in a .py module. Question 2 needs to be completed in a .html file as explained during the lectures. Once finished, compress (zip) both files with the following format: u123456_FinalAssignment.zip. Upload your file in Canvas using the submission drop-box provided in module “Final Assignment”. Note that only zip extensions are allowed to be submitted.

## Question 1: [25 marks]

Write a robust program that requests an integer from 1 through 10 and calculates its reciprocal. Your program must perform according to the following sample output:



```
Enter an integer between 1 to 10: 11
You did not enter a number between 1 and 10!!!
Please, try again.

Enter an integer between 1 to 10: ten
You did not enter an integer!!!
Please, try again.

Enter an integer between 1 to 10: .3
You did not enter an integer!!!
Please, try again.

Enter an integer between 1 to 10: 0
Oops, you entered zero.
Please, try again.

Enter an integer between 1 to 10: 3
The Reciprocal of your number is 0.3333333333333333.
```

**Business rules:**

- The exception handling must be done using 'try' statement. No other way is accepted (for instance, using if-elif-else). That is, no 'try', no marks.

- The program must keep running until the user enters a number between one and ten (inclusive) and receives the value of the reciprocal in the screen.
- Reciprocal is calculated as 1 divided by the number. For example, the reciprocal of 3 is 1/3=0.33.
- Markers will test your code using various examples, including the ones given above.

**Marks' allocation:**

- Program asks the user to enter a number. [3 marks]
- Program handles entering whole numbers between 1 and 10 inclusive. [3 marks]
- Program handles user entering numbers outside requested interval. [4 marks]
- Program handles user entering strings. [3 marks]
- Program handles user entering floats. [3 marks]
- Program handles user entering a zero. [3 marks]
- Program calculates reciprocal. [3 marks]
- Program keeps running unless the user enters requested input. [3 marks]

## Question 2 [25 marks]

You are asked to work out the required tests needed to assess the performance of a function called trip_OK(). The description of the problem, business rules, variables and variable types (including a test example) are included in the file "Holidays.html" located in the same section of this document on Canvas.

You need to edit this HTML file to include the test cases as described during the lectures and tutorials. All pieces of software to test are designed such that they do not need more than 10 tests. That is, you only have up to 10 tests available to assess all boundaries and partitions.

Be sure to save your answers and submit your html file as described in the Section "Submission" of this document.

**Marks' allocation:**

- All boundaries are tested. [8 marks]
- All partitions are tested. [8 marks]
- Proper descriptions are given for every test. [9 marks]
- You may need at least six tests to check all partitions and boundaries. No more than 10 tests are allowed. That is, all tests above the 10th won't be considered for marking purposes.