

# ENGG1003 - Thursday Week 3

Review of Monday's lecture  
& overview of Week 4 assessed lab

Steve Weller

University of Newcastle

11 March, 2021

Last compiled: March 11, 2021 3:32pm +11:00

# Lecture overview

## 1 Review of Monday's lecture

- ▶ `if-elif-else`
- ▶ `for` loop
- ▶ `while` loop

## 2 Overview of Week 4 assessed lab

- ▶ review of main topics covered in weeks 1–3
- ▶ what to expect in assessed lab
- ▶ reminder: worth 5% of overall course grade

# 1) Review of Monday's lecture

Reminder of main ideas in Monday's lecture:

- Branching: **if**, **elif** and **else**
  - ▶ conditional execution of code blocks
    - `if`
    - `if-else`
    - `if-elif-else`
- Iteration using **for** loop
  - ▶ fixed number of iterations
- Iteration using **while**
  - ▶ keep iterating whenever a Boolean condition is satisfied

# Branching: `if`

## Example 1

Write a Python program which takes an integer  $N \in \{1, 2, 3, 4, 5, 6\}$  and displays a message as follows:

- $N = 1$ , display “You win a prize!”
- $N \in \{2, 3, 4, 5, 6\}$ , no message is displayed
- Live demo

# Python code and output for Example 1

```
N = 1
if N==1:
    print('You win a prize!')
```

if N==1

Week3Thu\_if ×

"C:\Users\srw245\Documents\teaching  
You win a prize!

Process finished with exit code 0

# Branching: `if-elif-else`


## Example 2

Write a Python program which takes an integer  $N \in \{1, 2, 3, 4, 5, 6\}$  and displays a message as follows:

- $N = 1$ , display “You win first prize!”
- $N = 2$ , display “You win second prize!”
- $N \in \{3, 4, 5, 6\}$ , display “Sorry, no prize”
- Live demo

# Python code and output for Example 2

```
N = 1
if N==1:
    print('You win first prize!')
elif N==2:
    print('You win second prize!')
else:
    print('Sorry, no prize')
```

 Week3Thu\_ifelif ×

```
"C:\Users\srw245\Documents\teachi
You win first prize!
```

```
Process finished with exit code 0
```

# Iteration using `for` loop

## Example 3

Write a Python program which uses a **`for`** loop to print the numbers 1, 2, 3, ..., 10 to the console.

Your program should use the basic form of `for` loop

- ie: in loop header, use  
`[1, 2, 3, 4, 5, 6, 7, 8, 9, 10]`
- Live demo



# Python code and output for Example 3

```
for i in [1,2,3,4,5,6,7,8,9,10]:  
    print(i)
```

for i in [1,2,3,4,5,6,7,8,9,10]

Week3ThuEx3 x

"C:\Users\srw245\Documents\teach

1  
2  
3  
4  
5  
6  
7  
8  
9  
10

## Example 4

Modify your solution to Example 3 to use a **for** loop to print the *squares* of the numbers 1, 2, 3, ..., 10 to the console using the following display format:

$$1 ** 2 = 1$$

$$2 ** 2 = 4$$

$$3 ** 2 = 9$$

$$4 ** 2 = 16$$

.

.

.

$$10 ** 2 = 100$$

- Live demo

# Python code and output for Example 4

```
for i in [1,2,3,4,5,6,7,8,9,10]:  
    print('{}**2 = {}'.format(i,i**2))  
  
for i in [1,2,3,4,5,6,7,8,9,10]
```

Week3ThuEx4 x

"C:\Users\srw245\Documents\teaching so

```
1**2 = 1  
2**2 = 4  
3**2 = 9  
4**2 = 16  
5**2 = 25  
6**2 = 36  
7**2 = 49  
8**2 = 64  
9**2 = 81  
10**2 = 100
```

## Example 5

Modify your solution to Example 4 to use a **for** loop to print the squares of the numbers 1, 2, 3, ..., 1000 to the console using the display format in Example 4. You *must* use the `range` function in the `for` loop header.

`1**2 = 1`

`2**2 = 4`

`3**2 = 9`

`4**2 = 16`

`...`

`1000**2 = 1000000`

- Live demo

# Python code and output for Example 5

```
for i in range(1,1001,1):  
    print('{}**2 = {}'.format(i,i**2))
```

```
994**2 = 988036  
995**2 = 990025  
996**2 = 992016  
997**2 = 994009  
998**2 = 996004  
999**2 = 998001  
1000**2 = 1000000
```

# Iteration using `while` loop

## Example 6

Write a Python program which calculates the sum of the squares of the first  $N$  integers:

$$S_N = 1^2 + 2^2 + 3^2 + \cdots + N^2$$

- your program *must* use a `while` loop
- demonstrate your program for  $N = 10$
- check your answer is correct using the formula

$$S_N = \frac{N(N+1)(2N+1)}{6}$$

# How do we start to solve this problem?

- Don't try and solve the whole problem at once!
- Start small and build up your code one small step at a time
  - ▶ ... testing along the way
- Suggested first step: display 1, 2, 3, ..., 10 to the console
  - ▶ tests that your `while` loop syntax is correct
- Coding in small steps often helps in better understanding the problem

# Python code and output for Example 6

```
N = 20;
i = 1
sum = 0
while i <= N:
    print(i**2)
    sum = sum + i**2
    i = i + 1

print('sum = {}'.format(sum))
print('S_N = {}'.format(N*(N+1)*(2*N+1)/6))
```

```
196
225
256
289
324
361
400
sum = 2870
S_N = 2870.0
```



## 2) Overview of assessed lab in week 4

Key topics we've covered in the course so far:

- 1 Python program with library function
- 2 Simple plotting
- 3 Simple printing
- 4 Arrays
- 5 Branching: `if-elif-else`
- 6 Looping: `for` and `while`

# Assessed lab in week 4

- Each student assessed in the face-face lab sessions
- Worth 5% towards course grade
- 5 marks in lab sheet
- Two questions
- Answer both questions
- Sample paper will shortly be in BB > course materials > week 3

# Assessed lab in week 4

## 1 Instructions

This document has TWO tasks. Attempt both tasks.

**Start time:** 10 minutes after the timetabled lab time

**Time Allowed:** 1 hour

This is an OPEN BOOK assessment task. You may bring any paper-based material into the lab and access any PRE-EXISTING Internet resource.

The following, however, are prohibited:

- No use of mobile phones
- No use of Instant messaging
- No talking to peers
- No Help from demonstrators
- No reading from other's screens
- No active participation in online forums