Welcome to COMPSCI 101
Principles of Programming
Lecture 1 - Introduction

Learning outcomes

- At the end of this lecture, students should be able to understand:
  - where to obtain information about COMPSCI 101
  - which parts of the COMPSCI 101 assessment contribute to the practical mark
  - which parts of the COMPSCI 101 assessment contribute to the theory mark
  - that to pass COMPSCI 101, both the practical part of the course and the theory part of the course need to be passed
  - an algorithm

We are using Canvas

- We will be using the Canvas learning management system this semester.
  [https://canvas.auckland.ac.nz](https://canvas.auckland.ac.nz)

Canvas

- On Canvas you will find:
  - your marks
  - class announcements
  - lecture recordings
  - link to the COMPSCI 101 website
The COMPSCI 101 website

- The COMPSCI 101 website can be reached by logging onto the Canvas website:  
  https://canvas.auckland.ac.nz
- or by going directly to the COMPSCI 101 website:  
  https://www.cs.auckland.ac.nz/courses/compsci101s1c/

- Here you will find all the information about our course set up.
- Get used to looking at this website for information about lecture slides, lab documents, assignment resources, assessment, people involved in the course and lots more.

People in this course

Ann Cameron (Lab Supervisor)
Email: ann@cs.auckland.ac.nz
Room: 303.413

People in this course

Damir Azhar (Coordinator)
Email: damir.azhar@auckland.ac.nz
Room: 303.411

People in this course

Adriana Ferraro
Email: adriana@cs.auckland.ac.nz
Room: 303.415
People in this course

Angela Chang
Email: angela@cs.auckland.ac.nz
Room: 303.414

Jing Sun
Email: j.sun@cs.auckland.ac.nz
Room: 303.522

Office Hours

- Open door policy – Visit any time

Lecture schedule

- On the course information sheet there is a schedule of the lectures for COMPSCI 101.
Lecture Slides

- Lecture slides will be available on the web before each lecture.

https://www.cs.auckland.ac.nz/courses/compsci101s1c/lectures

There is no textbook for CompSci 101

- There is no textbook but we do have an online reference book, Think Python – How to think like a computer scientist.

- Please be aware that we are teaching the COMPSCI 101 material in a different order to the ordering in this book. This book is a reference book, not a textbook for this course.

https://www.cs.auckland.ac.nz/courses/compsci101s1c/resources/

Labs

- Labs start on the second week: March 13th – March 17th

- You must attend one 2 hour tutorial lab session each week.
- You will have enrolled in your lab time through Student Services Online. You should attend the same lab times each week.
- Labs are held in room 279 (Building 303S) which is on the second floor of the Computer Science building.

https://www.cs.auckland.ac.nz/courses/compsci101s1c/labs/

More about labs

- Labs start on the second week: March 13th – March 17th
- There are 9 labs and each lab is worth 1% of your final mark.
- At your lab time you will be given programming problems to solve within the 2 hours for your lab.

https://www.cs.auckland.ac.nz/courses/compsci101s1c/labs/
**Before the first lab**

- Visit the lab this week.
- Before the first lab you need to complete the lab preparation sheet (I will hand this out today).
- In order to fill the sheet you need to visit the COMPSCI 101 lab sometime this week.

**Assignments**

- The assignments are worth 11% of your final mark.
- Assignments give you the experience of solving problems on your own. Never share your code.

There are 5 assignments in total worth 11% of your final mark. For three of these five assignments (7% of your final mark), you are required to write and submit one or more programs.

All assignments are due at 4:30pm on the due date.

Three of the five assignments are handed in using the Assignment Drop Box:

https://adb.auckland.ac.nz/Home/

There are 5 assignments in total worth 11% of your final mark. For two of these five assignments (a total of 4%), you are required to use CodeRunner.

The CodeRunner tool is designed to help you practise by presenting you with a set of coding and other exercises. Students can work with on-line exercises using the Moodle learning system.

https://www.coderunner.auckland.ac.nz/moodle/

All assignments are due at 4:30pm on the due date.

Information about using CodeRunner is available on COMPSCI 101 assignments web page:

https://www.cs.auckland.ac.nz/courses/compsci101s1c/assignments/
Plagiarism

Plagiarism: Any work that you take credit for, but which is done by someone else. This is treated very seriously in an academic environment.

Policy
- All assignments will be checked for copying
- Everyone involved is penalised
- Disciplinary action will be taken in all cases of plagiarism

Advice
- Don't ever copy an assignment (or part of an assignment) from anyone
- Don't ever allow anyone to copy your assignment

https://www.academicintegrity.auckland.ac.nz

Test

- The test is worth 15% of your final mark
- The test date and time is:
  Wednesday 12th April 6:30pm - 7:45pm
- More about this closer to the time.

Email Damir Azhar (damir.azhar@auckland.ac.nz) the COMPSCI 101 course coordinator, if you are unable to attend the test.

Exam

- The exam is worth 65% of your final mark.
- More about this closer to the time.

Passing COMPSCI 101

Assessment

<table>
<thead>
<tr>
<th>Practical</th>
<th>Theory</th>
</tr>
</thead>
<tbody>
<tr>
<td>Labs</td>
<td>9%</td>
</tr>
<tr>
<td>Assignments</td>
<td>11%</td>
</tr>
<tr>
<td>Test</td>
<td>15%</td>
</tr>
<tr>
<td>Exam</td>
<td>65%</td>
</tr>
</tbody>
</table>

To pass the course, you MUST pass the PRACTICAL (i.e. get 10 / 20 or more) and you MUST pass the THEORY (i.e. get 40 / 80 or more).

Learning outcomes for COMPSCI 101

- determine the state of the program both during and after execution, given a code listing that may include functions and parameters, loops, conditionals and sequences,
- implement a given algorithm using Python,
- show that a program meets given specifications by writing appropriate tests,
- provide a useful level of documentation, in the form of program comments, for all programs developed,
- decompose a simple problem into several smaller tasks,
- given a brief textual description of the problem,
- compose functions that perform a specified task into a program that solves a given problem,
- describe program design and syntax using written language.

Algorithms

- A finite set of steps that specify a sequence of operations to be carried out in order to solve a specific problem.

A better definition:

An algorithm is a well-defined, unambiguous sequence of steps.

http://www.webopedia.com/TERM/A/algorithm.html

Computing resources

Undergraduate Labs:
There are demonstrators in these labs to help you.

GCL (room 091) – Ground Floor Computer Lab
This is a quiet lab.
Algorithms – what kind of steps?
An algorithm is a well-defined, unambiguous sequence of steps

HOW TO GET TO UNI

Walk to the bus stop at the shops up the road
Get on bus number "101"
Pay the bus driver $4.50
Get off at the Symonds St bus stop
Walk 200m to the Computer Science building

HOW TO WALK TO THE BUS STOP

Open the front door
IF it is raining THEN take an umbrella
Walk down the driveway and turn left
Walk 50m down the street

HOW TO PAY THE BUS FARE

Open wallet
WHILE you still haven't paid enough
   give the driver another coin
Take a seat

Basic programming steps

TASK

1 design

ALGORITHM

CODE

2 coding

PROGRAM

3 testing
Programming - Step 2 - write the code

We will use the **Python programming language** to implement our algorithms.

---

**ALGORITHM**

2 coding

---

```
***************
|   |
|   |
|   |

Word: _ _ _ _ _ _ _ _ _
Letters Missed: r m k c u d
Letters Guessed: a e s n o l b
Letters Available: fghijpqtvwxyz

Enter a letter: a

---

/|
/ \|

Word: l a _ _ b o n e s
Letters Missed: r m k c u d
Letters Guessed: a e s n o l b
Letters Available: fghijpqtvwxyz

Sorry, you have lost in the game of **Hangman**...

---

X enter your move? (1-9): 5

---

- O
- X

Computer move

***************
X - O
O X X
X O O

X enter your move? (1-9): 2

***************
X X O
O X X
X O O

Computer move

***************
X X O
O X X
X O O

X enter your next move? (1-9): 1

---

The result is a draw.

---

**Hangman**

Enter a letter: d

---

0
/ \|

Word: l a _ _ b o n e s
Letters Missed: r m k c u d
Letters Guessed: a e s n o l b
Letters Available: fghijpqtvwxyz

Enter a letter: a

---

The word was lazybones

---

**TicTacToe**

```
***************
| X |
| O |
|   |

X enter your move? (1-9): 5

---

- O
- X

Computer move

***************
X - O
O X X
X O O

X enter your move? (1-9): 2

***************
X X O
O X X
X O O

Computer move

***************
X X O
O X X
X O O

X enter your next move? (1-9): 1

---

The result is a draw.

---

**Hangman algorithm**

(1) generate a secret word
(2) display a "_" for each character in the word
(3) set "misses" to 0
(4) wait for the user to guess a letter
(5) IF the letter is in the word
   replace the corresponding "_" with the letter
   otherwise
   add 1 to "misses"
(6) IF the whole word is guessed THEN
   display "Well Done"
(7) IF the user presses the UP key THEN
(8) generate a secret word
(9) display a "_" for each character in the word
(10) set "misses" to 0
(11) GO BACK to (4)
Hangman algorithm

A  display a "_" for each character in the word
B  IF the letter is in the word
   replace the corresponding "_" with the letter
   otherwise
   add 1 to "misses"
C  generate a secret word
D  IF the user presses the UP key (i.e. get a new word) THEN
E  wait for the user to guess a letter
F  generate a secret word
G  IF the whole word is guessed THEN
   display "Well Done"
H  set "misses" to 0
I  GO BACK to (Wait for the user to guess a letter)
J  set "misses" to 0
K  generate a secret word

Plagiarism

What is plagiarism: Any work that you take credit for, but which is
done by someone else. This is treated very seriously in an
academic environment.

Policy

Everyone involved in copying is penalised
Disciplinary action may be taken

Advice

Don’t ever copy an assignment (or part of an
assignment) from anyone
Don’t ever allow anyone to copy your assignment