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
- understand which parts of the COMPSCI 101 assessment contribute to the practical mark
- understand which parts of the COMPSCI 101 assessment contribute to the theory mark
- understand that to pass COMPSCI 101, both the practical part of the course and the theory part of the course need to be passed
- understand an algorithm
We are using Canvas

We will be using the Canvas Learning Management system this semester. All the material on the COMPSCI 101 website can be accessed through Canvas:

https://canvas.auckland.ac.nz
As well as using Canvas, COMPSCI 101 has a course website.

The COMPSCI 101 website can be reached by logging onto the Canvas website,

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.
The COMPSCI 101 website

Most of the resources you will need (e.g., lecture slides, lab material, assignments, course information) can be found on the COMPSCI 101 website:

https://www.cs.auckland.ac.nz/courses/compsci101s1c/
People in this course

Ann Cameron (Lab Supervisor)
Email: ann@cs.auckland.ac.nz
Phone: 9234947
Room: 303.413
People in this course

**Damir Azhar (Course Coordinator)**

Email: damir.azhar@auckland.ac.nz
Room: 303.411
People in this course

Adriana Ferraro

Email: adriana@cs.auckland.ac.nz
Phone: 9237113
Room: 303.415
People in this course

Angela Chang

Email: a.chang@auckland.ac.nz
Phone: 9236620
Room: 303.413
People in this course

Jing Sun

Email: j.sun@cs.auckland.ac.nz
Room: 303.522
Office Hours

- Open door policy – Visit any time
On the course information sheet there is a schedule of the lectures for COMPSCI 101.

This course is an introduction to programming computer science, a general course in the Computer Science department and is taken by students from all disciplines wishing to have an understanding of computing. By the end of this course, students will have a knowledge and understanding of the detailed requirements of a professional programmer, and will be able to write programs that are well structured, correct, easy to read and test. This course is intended for students who have not previously studied programming. Students also need to develop an understanding of how to use and manipulate data and algorithms. Students also need to develop the ability to write and develop and testing programs.

The course covers simple variables, expressions, input and output, functions, using standard data structures such as lists and standard Python modules.
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 in the second week: March 11 - March 15

- You must attend one 2 hour tutorial lab sessions each week. You will have enrolled in your lab time through Student Services Online. You should attend the same lab time 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 in the second week: March 11 - March 15
- There are 9 labs worth 13.5% 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 on **Thursday** between 3pm and 4pm.
- Before the first lab you need to complete the lab preparation sheet (I will hand this out).
- In order to fill the sheet you need to visit the COMPSCI 101 lab on Thursday. Immediately after the lecture today I will be taking people across to the COMPSCI 101 lab.
Assignments

- The assignments are worth 16.5% of your final mark.
- Assignments give you the experience of solving problems on your own.
- Never share your code.

Solve the problem on your own – discuss the assignment with others but never share code.

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

- There are 5 assignments in total worth 16.5% of your final mark.
- All assignments are due at 4:30pm on the due date.
- For three of these five assignments (10.5% of your final mark), you are required to write and submit one or more programs.
- Three of the five assignments are handed in using the Assignment Drop Box:

  ![Assignment Drop Box](https://adb.auckland.ac.nz/Home)
Assignments - CodeRunner 2

- There are 5 assignments in total worth 16.5% of your final mark.
- All assignments are due at 4:30pm on the due date.
- For two of these five assignments (a total of 6%), you are required to use CodeRunner 2.
- The CodeRunner 2 tool is designed to help you practice 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.coderunner2.auckland.ac.nz/moodle/

- 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
Academic Integrity

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

- The test is worth 10% of your final mark
- The test date and time is:

  Monday 8\textsuperscript{th} April 6:30pm - 7:45pm

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

- The test is 75 minutes long plus 5 minutes of reading time. Arrive at 6:15 because your reading time starts at 6:25pm.
- More about this closer to the time.
Exam

- The exam is worth 60% of your final mark.

Information about missed exams, aegrotats, etc.

Illness, injury or other personal misfortune affecting exams

Apply if your personal circumstances are significantly affecting your exam performance or preparation.

Note: Written test aegrotat and compassionate applications are not yet available online and must be submitted in hard copy. For more information see: [Aegrotat or compassionate consideration for written tests](https://www.auckland.ac.nz/en/students/academic-information/exams-and-final-results/during-exams/aegrotat-and-compassionate-consideration.html).

Aegrotat and compassionate consideration applications for exams can be made online at [aegrotat.auckland.ac.nz/apply](https://www.auckland.ac.nz/en/students/academic-information/exams-and-final-results/during-exams/aegrotat-and-compassionate-consideration.html).
# Assessment

<table>
<thead>
<tr>
<th>PRACTICAL</th>
<th>THEORY</th>
</tr>
</thead>
<tbody>
<tr>
<td>Labs</td>
<td>Test</td>
</tr>
<tr>
<td>13.5%</td>
<td>10%</td>
</tr>
<tr>
<td>Assignments</td>
<td>Exam</td>
</tr>
<tr>
<td>16.5%</td>
<td>60%</td>
</tr>
</tbody>
</table>

**To pass the course**

you MUST pass the PRACTICAL (i.e., get 15 / 30 or more)

you MUST pass the THEORY (i.e., get 35 / 70 or more)
Canvas

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

https://canvas.auckland.ac.nz
Piazza

- Piazza is a Q&A web service integrated into Canvas.

- You can use Piazza to ask questions that the lecturers and your classmates can discuss and answer.

- Please never post your own code up on Piazza!
Computing resources

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

**GCL** (room G91) – Ground Floor Computer Lab
This is a *quiet* lab
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,
- compose functions that perform a specified task into a program that solves a given problem.
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.
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
An algorithm is a well-defined, *unambiguous* sequence of steps.

**HOW TO WALK TO THE BUS STOP**

1. Open the front door
2. **IF** it is raining **THEN** take an umbrella
3. Walk down the driveway and turn left
4. Walk 50m down the street
Algorithms - what kind of steps?

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

**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

1. design
2. coding
3. testing
We will use the **Python programming language** to implement our algorithms.