Introduction to Computer Programming
Lecture 1.1:

Course Introduction

Hemma Philamore

Department of Engineering Mathematics

We are …

Hemma Philamore
Unit Director

hemma.philamore@bristol.ac.uk

Mark Blyth
Head TA

m.blyth@bristol.ac.uk

Time Line


1
Oct 5th
2
Oct 12th
3
Oct 19th
4
Oct 26th
5
Nov 2nd
6
Reading Week
7
Nov 16th
8
Nov 23rd
9
Nov 30th
10
Dec 7th
11
Dec 14th
Nov 23rd: Coursework assignment set
Dec 11th: Coursework assignment deadline

Weekly Schedule

Monday:

- Watch recorded video tutorials (2-3 videos per week), Blackboard or YouTube: https://www.youtube.com/playlist?list=PLIePLwgNVGWElRQviZ2izI8UR8dYPWrTI
- Complete exercises by Friday of the same week

Wednesday (14:00-16:00):

- Optional drop-in class
- Support and help with completing exercises 1-to-1 support from teaching assistants via video chat

Friday (see timetable for your group meeting time):

- Group tutorial with teaching assistant

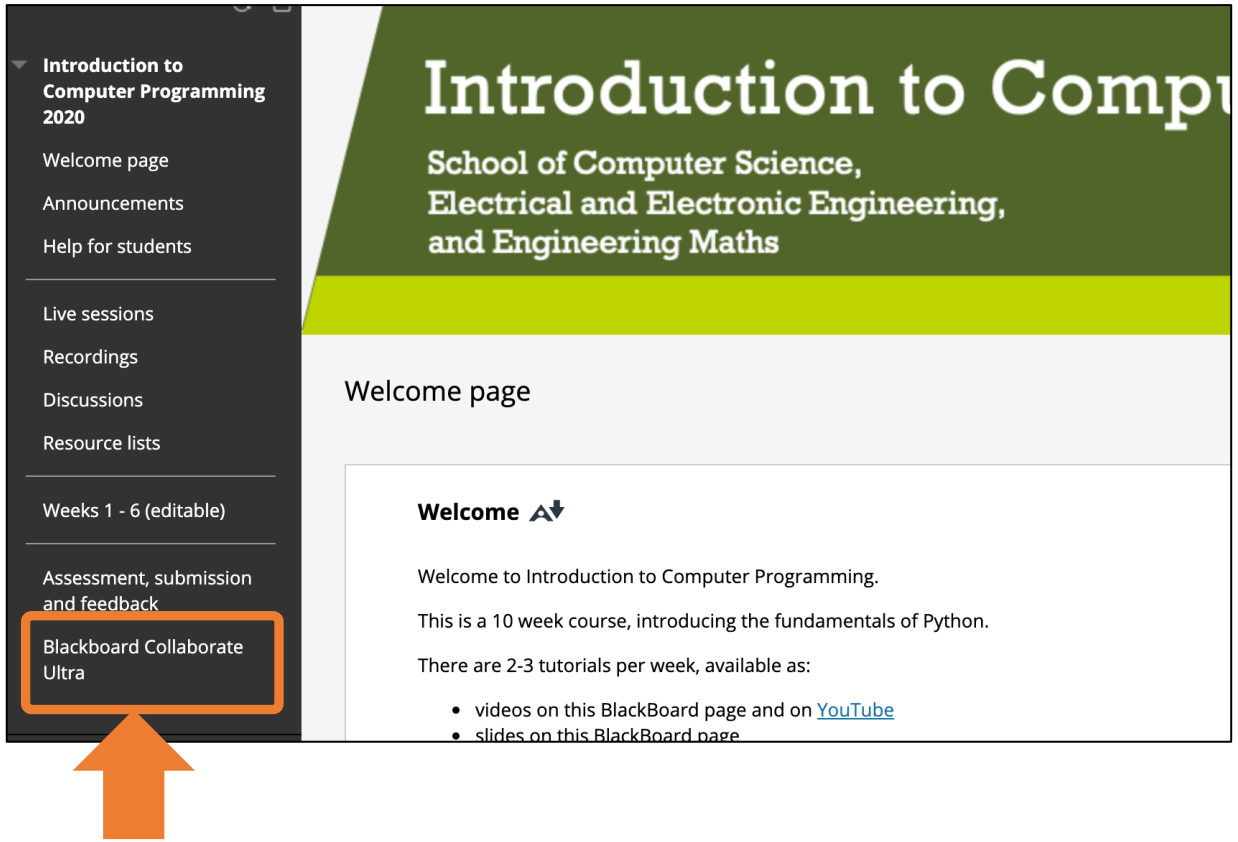
Blackboard site: http://www.ole.bris.ac.uk

- Course documents
  - Tutorial videos and slides
  - Examples
  - Weekly exercises
- Schedule
- Contact details

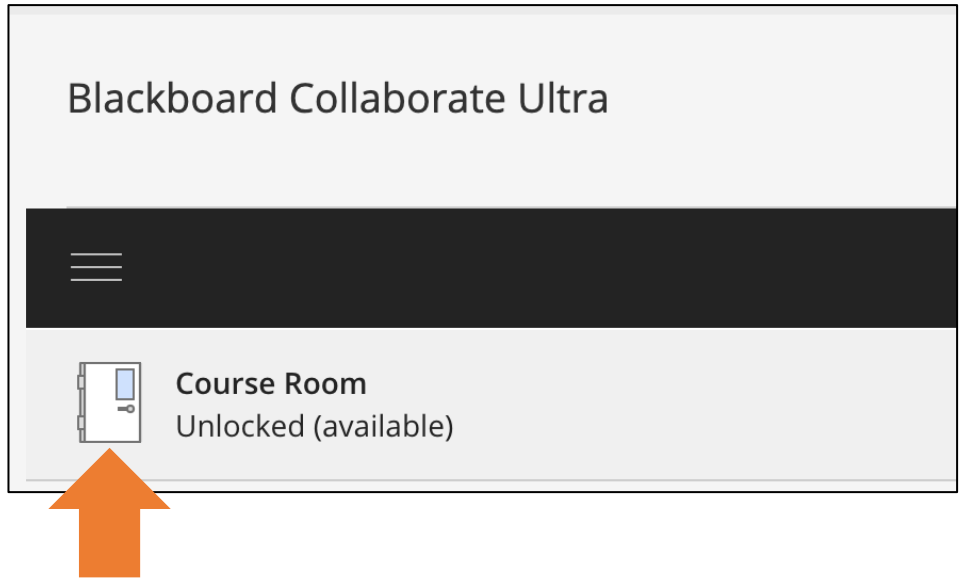
Drop-in Classes

Wednesday

1-to-1 support to help with weekly exercises



Choose Blackboard Collaborate Ultra from the side pane of the main Blackboard page



Click to enter the Course Room
You will be assigned a TA for 1-to-1 support

Tutorials

Friday

Schedule

10 teaching weeks (05/10/20 - 18/12/20)

[break: week 6, 09/11/20]

WEEKLY SCHEDULE

Monday:

- Watch recorded video tutorials (2-3 videos per week)
- Complete exercises by Friday of the same week

Wednesday (14:00-16:00):

- Optional drop-in class
- Support and help with completing exercises
- 1-to-1 support from teaching assistants via video chat

Friday (see timetable for your group meeting time):

- Group tutorial with teaching assistant

TEACHING ASSISTANT	TUTOR GROUP
MORNING (9:00-10:00)	
Abel Pacheco Ortega	1
Ercan Ezin	2
Euan Judd	3
Faegheh Sardari	4
Fanqi Zeng	5
Frederico Dias Paulino Da Costa	6
Huseyin Burak Akyol	7
Kit Simmonds	8
Perla Jazmin Mayo Diaz de Leon	9
Richard Pyle	10

TEACHING ASSISTANT	TUTOR GROUP
MORNING (9:00-10:00)	
Abel Pacheco Ortega	1
Ercan Ezin	2
Euan Judd	3
Faegheh Sardari	4
Fanqi Zeng	5
Frederico Dias Paulino Da Costa	6
Huseyin Burak Akyol	7
Kit Simmonds	8
Perla Jazmin Mayo Diaz de Leon	9
Richard Pyle	10
Yanan Liu	11
Zixuan Liu	12
Allison Horney	13
Holly Millea	14
Marceli Wac	15
Charles Khoury	16
Eddie Jones	17
Satya Rammohan& Charlie Leach	18

Morning 9:00-10:00
Afternoon 13:00-14:00

Your teaching assistant will email you to arrange your first tutorial.

Blackboard Collaborate Ultra

What do you need to know?

- **No** programming skills necessary!
- For those who know how to program, we'll make sure to keep it **interesting with extra challenges**
- Best way to learn programming is by doing it
- We are here to help you!

Weekly exercises are crucial!

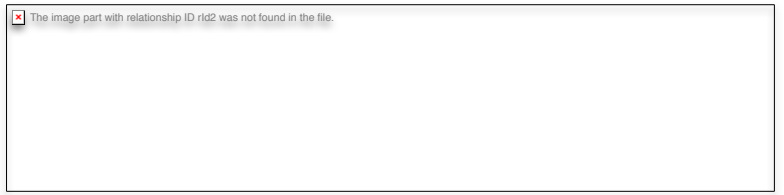
TAs are here to support you!

Use of personal laptops

- Remote learning – use of personal laptops for duration of course
- First task! → software installation (Anaconda, Spyder)
- Watch the tutorial videos (Blackboard or YouTube)
- We will help you as much as possible but you will all be using different computers and may experience different problems.

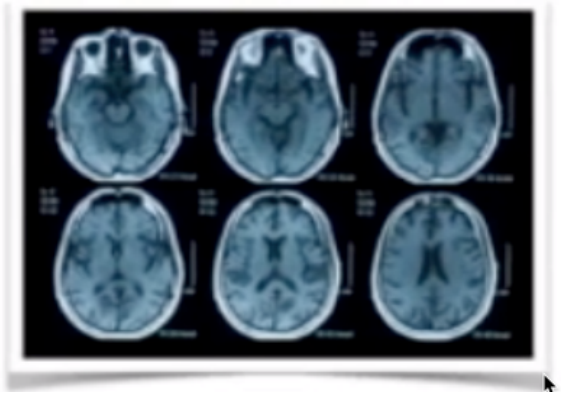
What is programming?

- Programming is telling a computer exactly what to do.
- You have to use a specific language.
- You tell the computer what to do in small steps → **algorithm**



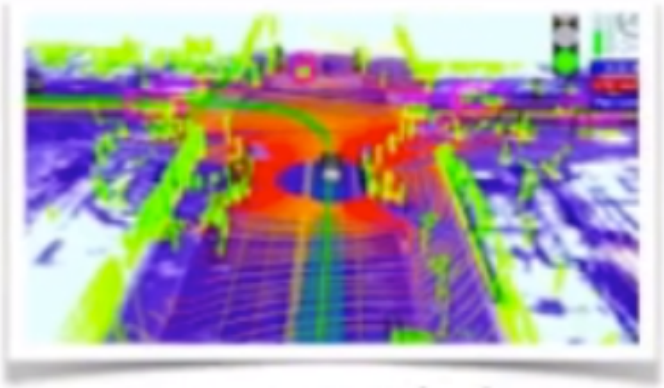
Why are we doing this course?

data science



artificial intelligence

social media



autonomous vehicles

What is an algorithm?

- **Algorithm:** self-contained step-by-step set of operations to be performed

- **Examples:**
  - Cooking recipe
  - How to solve Rubik's cube
  - Solve a quadratic equation
  - Find the highest value in the list
  - etc.

Think like a computer !

Example

Task: Lift operation

1. Wait until door is closed
2. Wait for button to be pressed
    - If button pressed is higher than current floor
        - Move lift upwards
    - If button pressed is lower than current floor
        - Move lift downwards
3. Wait until current floor equals button pressed
4. Open doors

Example

Task: Lift operation

1. Wait until door is closed
2. Wait for button to be pressed
    - If button pressed is higher than current floor
        - Move lift upwards
    - If button pressed is lower than current floor
        - Move lift downwards
3. Wait until current floor equals button pressed
4. Stop lift
5. Open doors

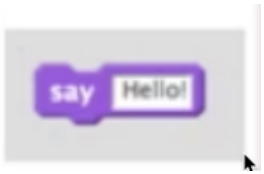
What languages does a computer understand?

1. Wait until door is closed
2. Wait for button to be pressed
    - If button pressed is higher than current floor
        - Move lift upwards
    - If button pressed is lower than current floor
        - Move lift downwards
3. Wait until current floor equals button pressed
4. Stop lift
5. Open doors

What languages does a computer understand?

our unit

Scratch



Python

```
print("Hello!")
```

C / Java

```
printf("Hello!");
println("Hello!");
```

machine code

```
push es
push ds
xor ax, ax
mov es, ax
mov ds, ax
…
```

binary code

```
10010011
10100111
00011010
…
```

High-level programming language

low-level programming language

What languages does a computer understand?

Python

print("Hello!")

✓ print("Hello!")

display("Hello!") ✗

printf("Hello!") ✗

print Hello! ✗

print (Hello!) ✗

Like every language it has rules called syntax.

Next steps

- ✓ Watch this week's tutorial videos
  - Software installation (Mac, Windows, Linux)
  - Spyder software
- ✓ Attempt this week's exercises
- ✓ Go to drop-in session if you have any problems *(Wednesday)*
- ✓ Attend group tutorial *(Friday)*