

COMP10020

Introduction to Programming II

Dr. Brian Mac Namee

brian.macnamee@ucd.ie

School of Computer Science

University College Dublin

COMP10020

TWO EXAMPLES

COMP10020

The Monty Hall Problem



COMP10020

The Big Deal



COMP10020

The Big Deal

1

2

3

- One door has a prize of a car behind it
- The other two doors have prizes of goats!

COMP10020

The Big Deal

1

2

3

- You have to pick a door to win the prize behind it

COMP10020

The Big Deal

1

2

3

- You have to pick a door to win the prize behind it
- But there is a twist!

COMP10020

The Big Deal

1

2

3

- After you pick a door, Monty reveals a goat behind one of the remaining doors

COMP10020

The Big Deal

1

2

3

- After you pick a door, Monty reveals a goat behind one of the remaining doors

COMP10020

The Big Deal

1

2

3

Let's have a go!

COMP10020

The Big Deal

1

2

3

- The big question is should you stick or switch?
 - Stick?
 - Switch?
 - No difference?

COMP10020

Write Some Code!

Jupyter Monty Hall Last Checkpoint: an hour ago (autosaved)

File Edit View Insert Cell Kernel Help Python [Root] O

Simulating The Monty Hall Problem

A function to play an instance of the Monty Hall problem

```
In [49]: import random
from __future__ import division

def play_monty_hall(strategy = "stick"):

    # Set up choice of three doors
    doors = [1,2,3]

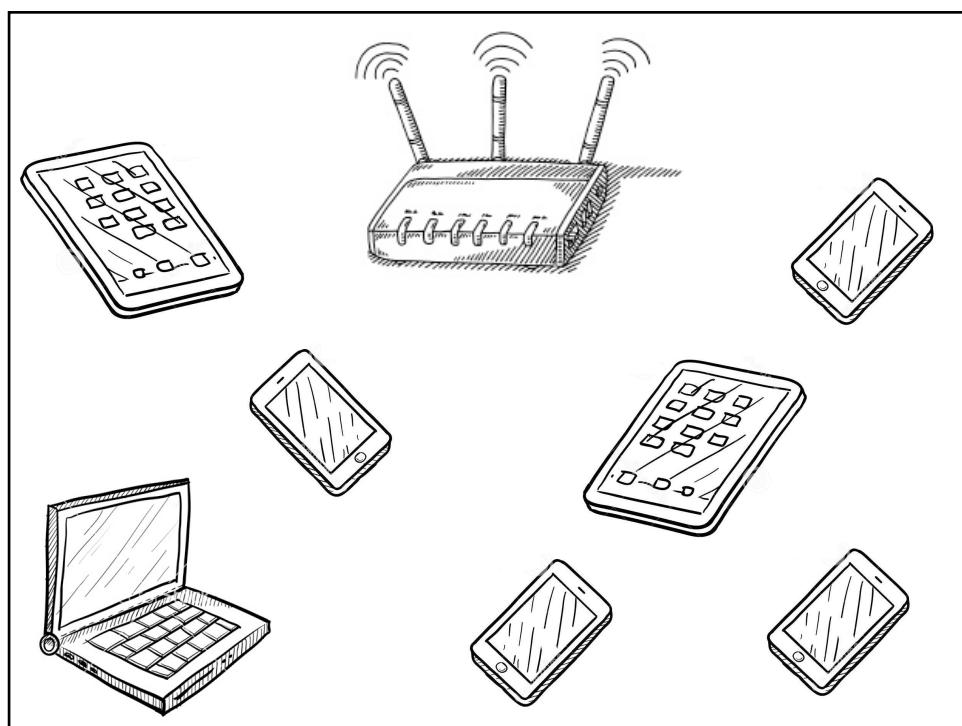
    # Randomly select the winning door for this round
    winning_door = random.choice(doors)

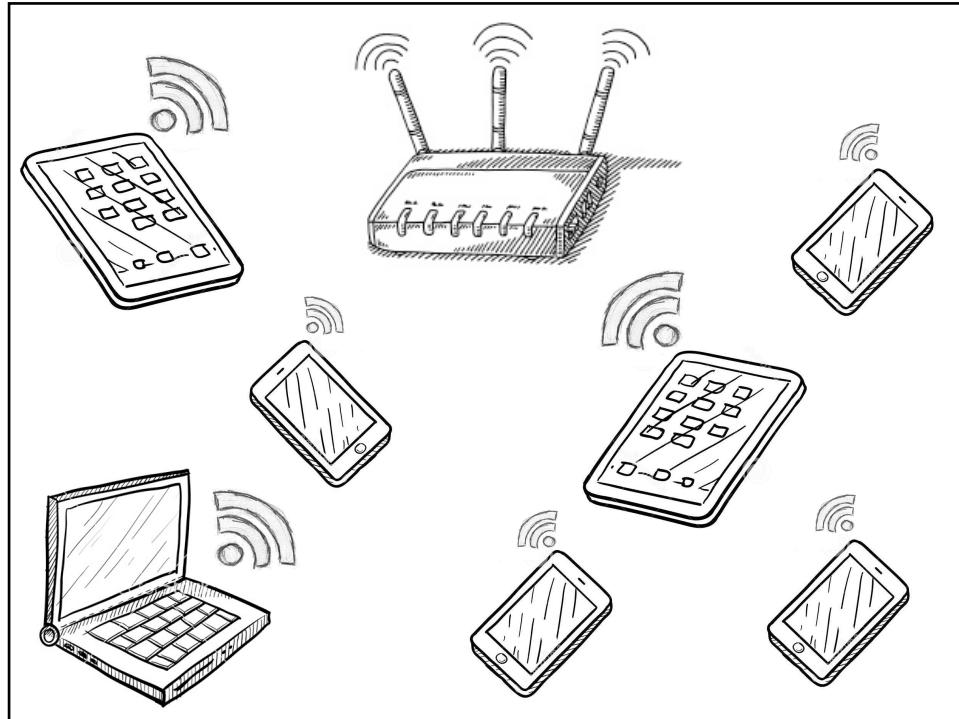
    # Randomly pick the player's door for this round
    player_door = random.choice(doors)

    # Randomly select one of the losing doors to reveal to the player
    temp_doors = list(doors)
    temp_doors.remove(player_door)
    if(winning_door != player_door):
        temp_doors.remove(winning_door)
    reveal_door = random.choice(temp_doors)

    # If the strategy is to switch after the reveal then do it!
    if(strategy == "switch"):
        temp_doors = list(doors)
        temp_doors.remove(reveal_door)
        temp_doors.remove(winning_door)
        player_door = temp_doors[0]
```

COMP10020





Write Some Code!

Jupyter Probe Request Sniffer Runnable Instructions Last Checkpoint: 09/22/2016 (unsaved changes)

```

if hasattr(pkt, 'info'):
    ssid = pkt.info.ssid
    ssid = ''.join(filter(lambda x: x in string.printable, ssid))
    # Not very good, but simple error handling - ssids of more than 32 characters probably indicate malef
    if(len(ssid) <= 32 and ssid.find("\x00") == -1):
        if(all(c in string.printable for c in ssid) and (len(ssid) > 5)):
            devices[pkt.addr2]['ssids'].add(ssid)

print(str(len(devices)) + " found")

```

33 found

Write out the devices read in into a nice csv format

```

In [8]: with open('devices' + str(time.time()) + '.csv', 'wb') as csvfile:
    writer = csv.writer(csvfile, delimiter=',', quotechar='|', quoting=csv.QUOTE_MINIMAL)
    writer.writerow(['ID', 'freq', 'manu', 'ssids', 'last_seen'])

    for d in devices:
        ssidString = ' | '.join(str(e) for e in devices[d]['ssids'])
        writer.writerow([d, devices[d]['freq'], devices[d]['manu'], ssidString, devices[d]['last_seen']])

```

Generate a list of the ssids found

```

In [9]: ssidsFound = []

# Iterate through the devices list
for d in devices:

    # get the list of ssids saved from a device
    ssids = devices[d]['ssids']

```

OVERVIEW

COMP10020

Module Outline

1. Python

- Revision
- Object oriented (OO) programming

2. Algorithms

- Problem solving
- Searching & ranking
- Data analysis

3. Data Science

- Data access
- Data manipulation
- Data analysis
- Data visualisation

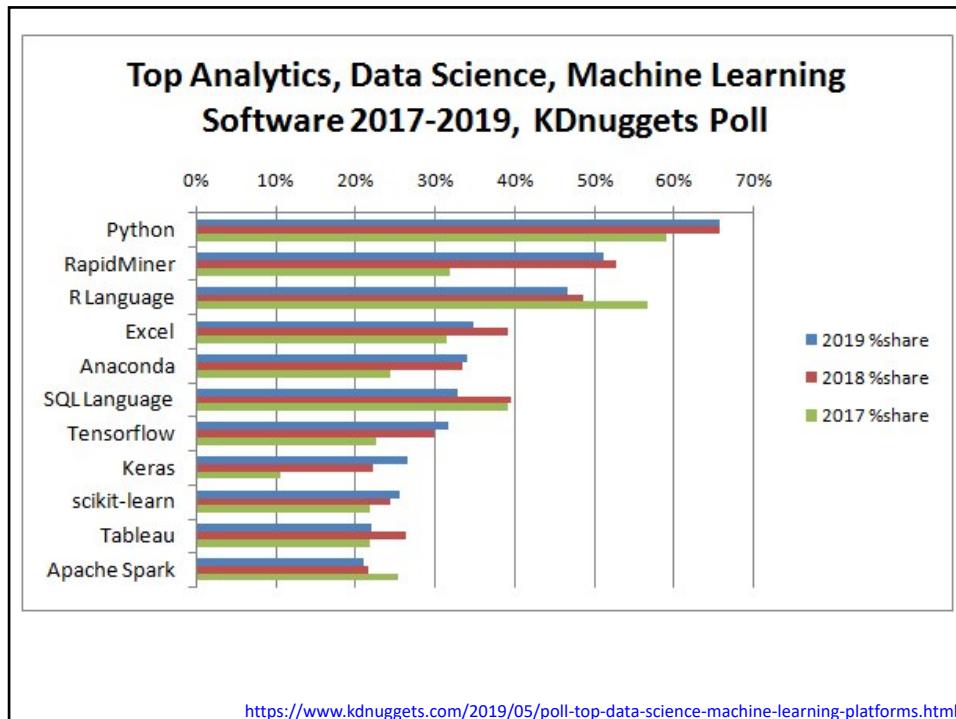
COMP10020

WHY PYTHON?

COMP10020

Rank	Language	Type	Score
1	Python	🌐💻⚙️	100.0
2	Java	🌐📱💻	96.3
3	C	📱💻⚙️	94.4
4	C++	📱💻⚙️	87.5
5	R	💻	81.5
6	JavaScript	🌐	79.4
7	C#	🌐📱💻⚙️	74.5
8	Matlab	💻	70.6
9	Swift	📱💻	69.1
10	Go	🌐💻	68.0

<https://spectrum.ieee.org/computing/software/the-top-programming-languages-2019>



Why Python?

Open source and well supported by freely available tools

Clean, concise, unambiguous syntax

Supports a variety of programming paradigms

- Simple scripts
- Object-oriented programming
- Interactive notebooks

Strong library support

Strong online community support

Why Python?

Open source and well supported by freely available tools

Clean

```
for line in open("file.txt"):
    for word in line.split():
        if
            word.endswith('ing'):
                print(word)
```

Support

–

–

–

Strong library support

Strong online community support

COMP10020

Why Python?

Open source and well supported by freely available tools

Clean, concise, unambiguous syntax

Supports a variety of programming paradigms

- Simple scripts
- Object-oriented programming
- Interactive notebooks

Strong library support

Strong online community support

COMP10020

Why Python?

Open source and well supported by freely available tools

Clean, concise code

Supports a wide range of applications

- Simple syntax
- Object-oriented
- Interactive

And you know it
already from
COMP10010!!

algorithms

Strong library support

Strong online community support

COMP10020

WHAT IS DATA SCIENCE?

COMP10020

What Is Data Science?

"data science" Search term + Add term

Interest over time

News headlines Forecast

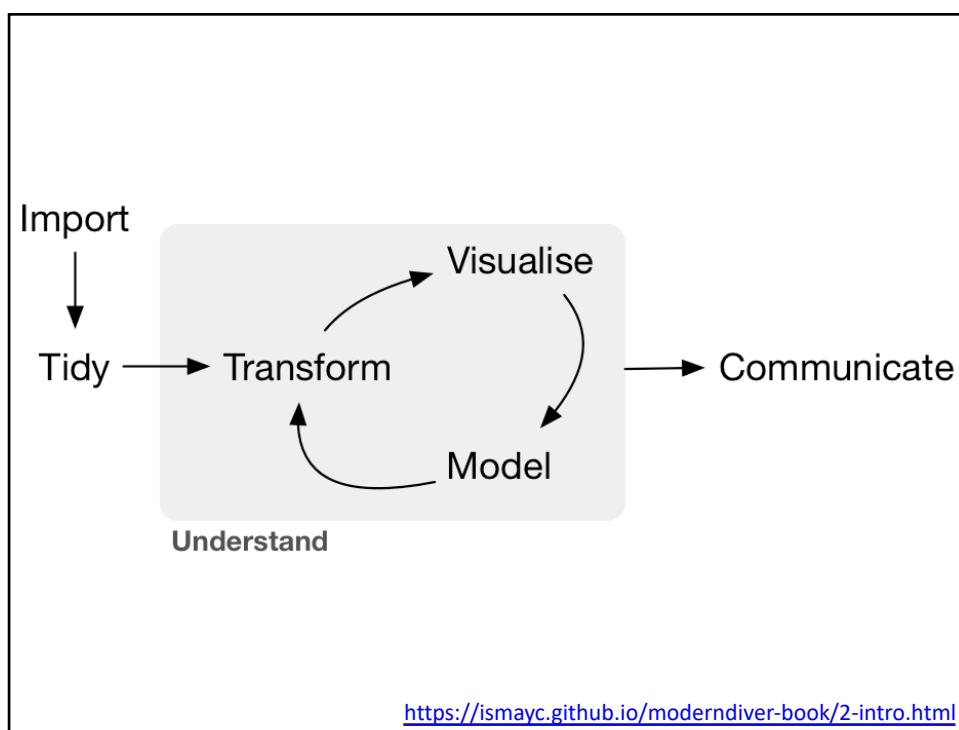
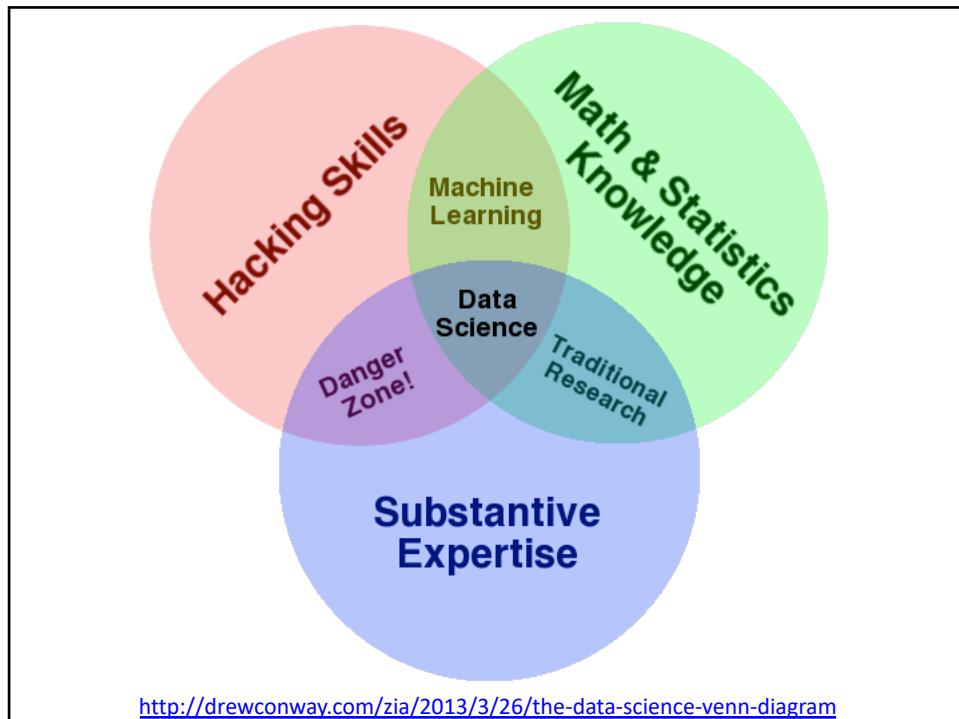
<https://www.google.com/trends/explore#q=%22data%20science%22&cmpt=q>

COMP10020

What Is Data Science?

At its core Data Science is about developing the infrastructure and processes for dealing with data at scale, recognising and understanding patterns within large, diverse datasets, generating predictions based on these patterns, and creating revealing visualizations and crafting compelling narratives with and about data

COMP10020



DATA-DRIVEN EVERYTHING



What Is Data Science?

Data Science brings together key ideas from multiple fields

- Computer science (algorithms, representation, visualization, application development)
- Statistics (modelling, analysis, prediction)
- Design (information design, interaction design)
- Psychology and cognitive science (language and perception),
- Humanities and social sciences (storytelling and narrative, social learning)

What Is Data Science?

Data Science brings together key ideas from multiple fields

- Computer science, visualisation
- Statistics
- Design
- Psychology, perception
- Humanities and social sciences (storytelling and narrative, social learning)

Data science is also a nice application area off which we can hang a programming course

COMP10020

ADMINISTRARIA

COMP10020

Practical Information

Lectures

- Tuesday 10:00 - 11:00 129 SCN
- Thursday 10:00 - 11:00 Th. O ART

Labs

- Tuesday 16:00 - 18:00 B108 ART
- Wednesday 16:00 - 18:00 B1.06 CSI
- Thursday 16:00 - 18:00 B108 ART

Materials

- All materials will be posted on Brightspace

COMP10020

Practical Information

Lectures

- Tuesday
- Thursday

Labs

- Tuesday
- Wednesday
- Thursday

Materials

- All materials will be posted on Brightspace

Labs do not start
until week 3

You will be assigned a
lab group

COMP10020

Assessment

Continuous assessment

- 60% of final mark
- A series of programming assignments in labs

End of semester exam

- 40% of final mark
- Exam covering practical and theoretical issues discussed in class

COMP10020

Plagiarism & UCD Computer Science

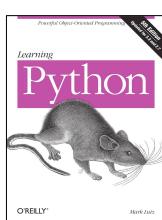
- **Plagiarism is a serious academic offence**
 - [Student Code, sections 6.2 & 6.3] or [UCD Registry Plagiarism Policy] or [CS Plagiarism policy and procedures]
- Our staff/demonstrators are **proactive** in looking for possible plagiarism
- Suspected plagiarism is investigated by the CS Plagiarism subcommittee
 - Usually includes an interview with student(s) involved
 - 1st offence: **usually** 0 or NG in the affected components
 - 2nd offence: more serious consequences e.g. UCD Disciplinary process
- Student who *enables* plagiarism is *equally responsible* for it
- **Examples** of plagiarism:
 - Copying the files of another student and submitting them as your own work
 - Copying some/all of an assignment from the Internet/book/etc without referencing it
 - Sharing images of your work with another student (by e-mail, FB messenger, WhatsApp, ...)
 - A group of students working on a solution, then individually submitting the same work
 - Students collaborating at too detailed a level e.g. consulting each other after implementing a line/block/segment of code and sharing the results

BOOKS & OTHER RESOURCES

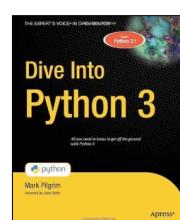
COMP10020

Books & Other Resources

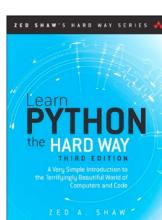
No specific textbook for this module



Learning Python
Mark Lutz
O'Reilly Media
www.learning-python.com/books/



Dive Into
Python 3
Mark Pilgrim
Apress
www.diveintopython3.net

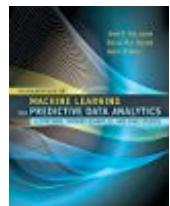


Learn Python the
Hard Way
Zed A. Shaw
www.learnpythontthehardway.org

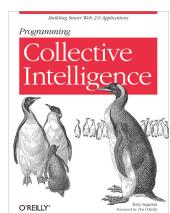
COMP10020

Books & Other Resources

No specific textbook for this module



Fundamentals of Machine Learning for
Predictive Data Analytics
John D. Kelleher, Brian Mac Namee,
Aoife D'Arcy
MIT Press
www.machinelearningbook.com



Programming Collective
Intelligence
Toby Segaran
O'Reilly Media
www.kiwitobes.com

COMP10020