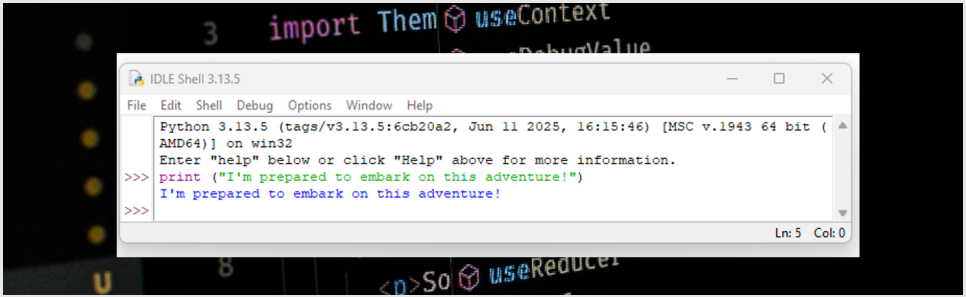
CSCM712: Welcome!

Betsy Chaparro-Rico & Cécilia Pradic

Swansea University
26/01/25



Who?

Cécilia Pradic (she/her)

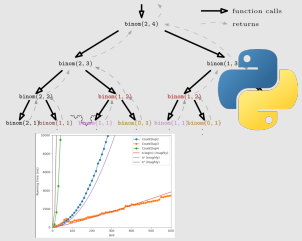
- Lecturer
- Research: logic, computability
- `c.pradic@swansea.ac.uk`
- Office: CoFo 410

Betsy Chaparro-Rico (she/her)

- Lecturer
- Research: robotics
- `b.d.m.chaparrorico@swansea.ac.uk`
- Office: CoFo 319

What?

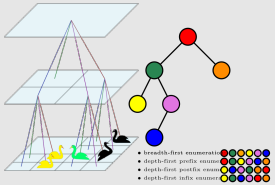
Introduction to python and algorithms



$\mathcal{O}(\log(n))$

- Lectures by Betsy
- Weeks 1-5

Algorithms & datastructures



- Lectures by Cécilia
- Weeks 6-10

What are we going to do?

- Introduction to Python
  - New language, but we expect you to pick it up quickly since you have previous experience with CSCM41
  - **Introduction** to algorithmics and datastructure

Aims

By the end of the module, you should be able to:

- know how to compute the time complexity of your code
- know some standard algorithms and datastructures
- be able to solve simple algorithmic problems on your own
- and apply that knowledge to your programming!

Should be clear to you by the end of the module

6. Concatenating immutable sequences always results in a new object. This means that building up a sequence by repeated concatenation will have a quadratic runtime cost in the total sequence length. To get a linear runtime cost, you must switch to one of the alternatives below:

   - if concatenating `str` objects, you can build a list and use

Is this a programming module?

In short, **no but yes**

- we will use python for motivation, examples and assessment
- but the focus won't be on learning python

But how do I benefit as a programmer?

- train to abstract away
- you will be able to write better more efficient code
- understand standard libraries and efficiency concerns

Some extrinsic motivation: job interview questions

Here are some typical job interview questions you should be able to answer after having taken the module:

- Find the largest element in an array
- What is a linked list?
- Calculate the length of a linked list
- Reverse a singly linked list
- Would you use a linked list or a dynamic array to store a list of actors in a video game? Why?
- Compute the sum of all the values in the nodes of a linked list
- Print the values in a singly linked list in reverse order
- Write insert and remove operations for a singly linked list
- How would you sort a singly linked list?
- What are the advantages and disadvantages of linked lists vs. arrays?

More interview questions

- What is a binary search tree?
- Implement an enumerator for binary tree traversal
- Write a function to search a BST for a key and return the found node
- When would you use a binary tree vs. a hash table?
- Write a function which will insert a number into a BST
- Write a function which will flatten a binary tree into an array
- Write a function which will link the siblings in a tree
- Write a method to set all the data in a tree nodes to the empty string.
- How fast can you compute the distance between two nodes in a tree-like graph?
- Implement a queue using two stacks

Lab

Goal

- Week 1-2: become proefficient in Python
- Week 3-10: tasks to ground your understanding of the theory

- Individual: don't cheat/share solutions
- Sign-off at the end of session/beginning of the next

Support

- your lab slot

Support

- your lab slot
- office hours:
  - Betsy: Monday 1:30-3:30pm, CoFo 319
  - Cécilia: Monday 2-4pm, CoFo 410

Support

- your lab slot
- office hours:
  - Betsy: Monday 1:30-3:30pm, CoFo 319
  - Cécilia: Monday 2-4pm, CoFo 410
- Preferrably engage in lectures, labs and office hours rather than email unless urgent or you have a compelling reason
- Computer science advisory sessions with tutors
  - Noon, entrance of the CoFo

Generative “AI” for this module?

No.
At least assuming you respect

- yourself
- your learning
- the people you are communicating with

Assessment

15% Labs

Individual, weekly labsheets with tasks to sign-off

Assessment

15% Labs

Individual, weekly labsheets with tasks to sign-off

15% coursework

Due March 11th. Pen-and-paper, on algorithmic complexity

Assessment

15% Labs

Individual, weekly labsheets with tasks to sign-off

15% coursework

Due March 11th. Pen-and-paper, on algorithmic complexity

70% Pen-and-paper exam in May

All topics in the module

- theory
- a few coding questions on datastructures

The textbook

We will be using a textbook for this module:

Problem Solving with Algorithms and Data Structures using Python
by Brad Miller and David Ranum

- Electronic version:
  `https://runestone.academy/ns/books/published/pythonds/`
- *We expect you to do some reading!*
  - We shall recommend sections to read after the lecture and before the lab.

The textbook

We will be using a textbook for this module:

Problem Solving with Algorithms and Data Structures using Python
by Brad Miller and David Ranum

- Electronic version:
  `https://runestone.academy/ns/books/published/pythonds/`
- *We expect you to do some reading!*
  - We shall recommend sections to read after the lecture and before the lab.
- That said, many sources exist that cover the material; we may mention some.
- Be wary/critical of sources of questionable quality

  (`geekforgeeks.com`, random post on fora such as `stackexchange`, ...

Questions

- Challenge: variety of backgrounds in the cohort
- Blessings: small class size

Communicate!

- Ask questions
- Ask to explain/revisit some points
- Give feedback & suggestions in the discussion board on canvas

Questions

- Challenge: variety of backgrounds in the cohort
- Blessings: small class size

Communicate!

- Ask questions
- Ask to explain/revisit some points
- Give feedback & suggestions in the discussion board on canvas

Any questions?

(before you actually get taught something by Betsy :))