ReCoDE – project-based cognitive apprenticeship learning for research computing and data science

John Pinney*, Jay DesLauriers*, Chris Cooling*,
Liam (Jianliang) Gao*, Jeremy Cohen**, Diego Alonso Alvarez***
and Katerina Michalickova*

*Research Computing and Data Science Programme (RCDS) at the Graduate School
**Department of Computing,
***Research Computing Service

IMPERIAL
Festival of Learning and Teaching 2024

Task:

- Complete PhD

Task:

- Complete PhD

Subtask:

- Learn computer programming

```
print( "Hello, world!" )
```

```
def fibonacci(n):
    if n in {0, 1}:
        return n

    prev, fib = 0, 1
    for _ in range(2, n + 1):
        prev, fib = fib, previous + fib

    return fib
```

function definition

parameter

if statement

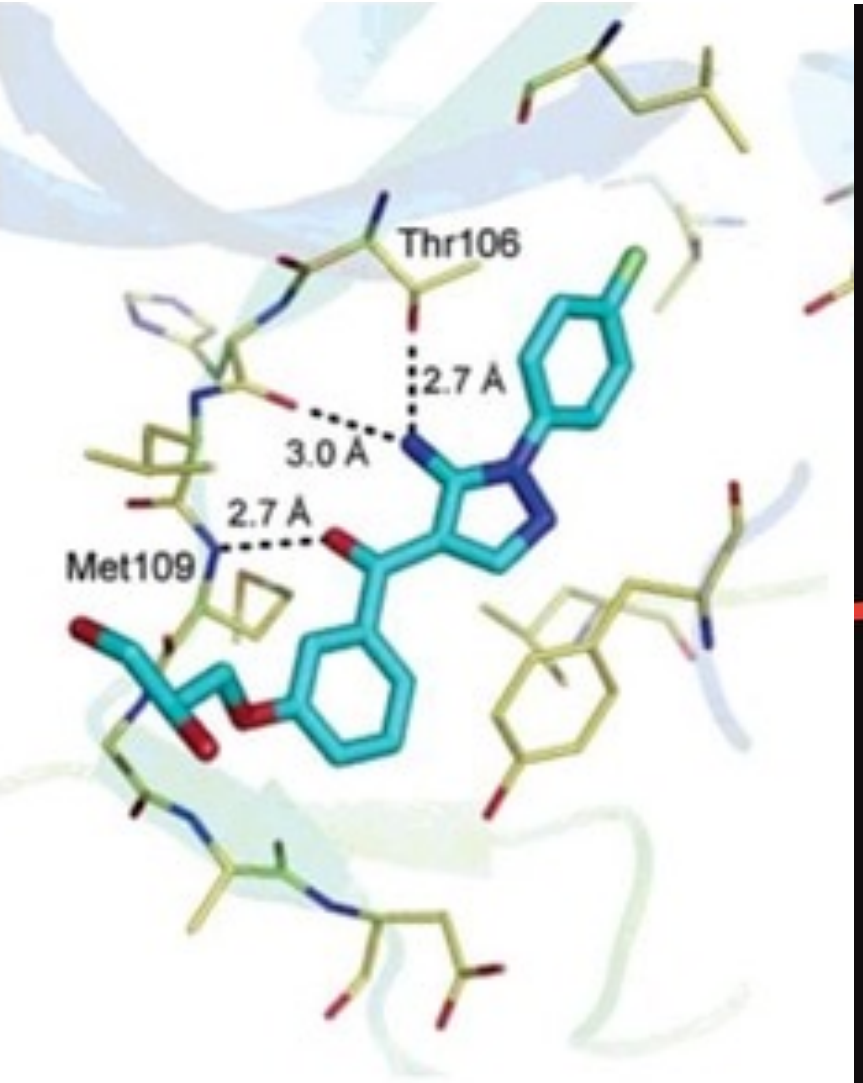
list

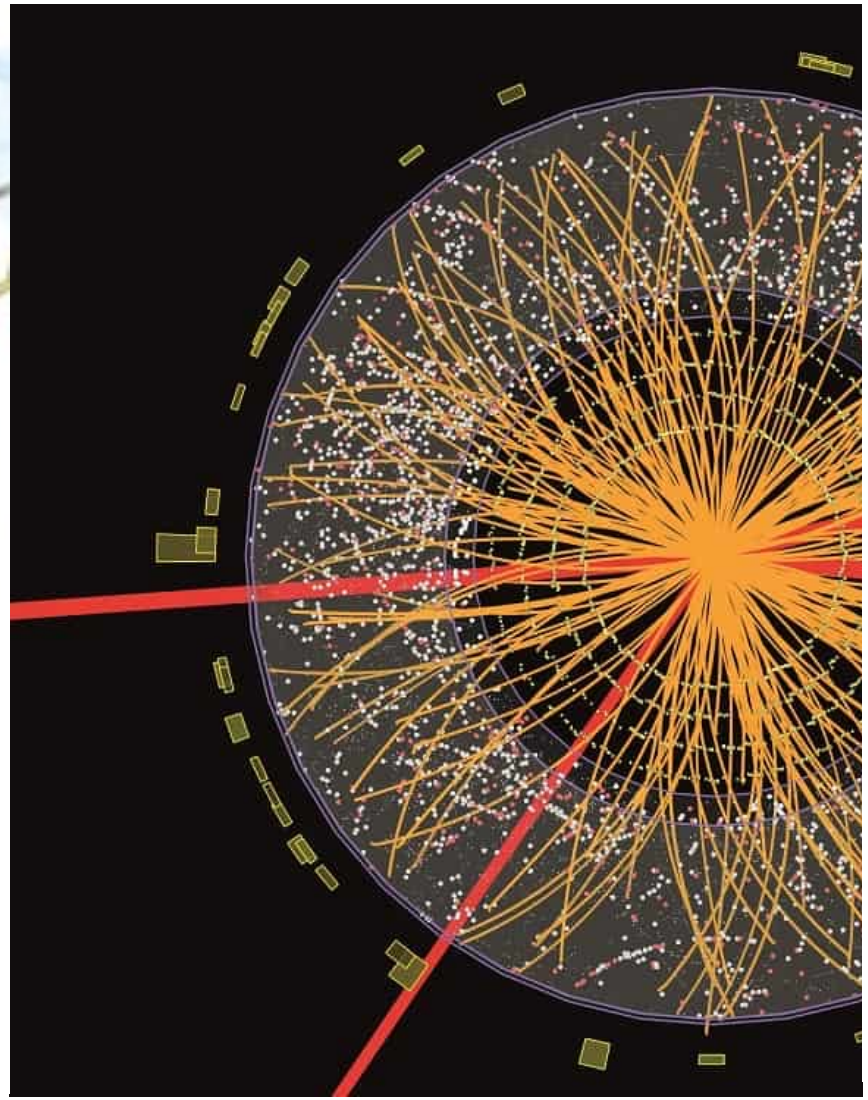
variable assignment

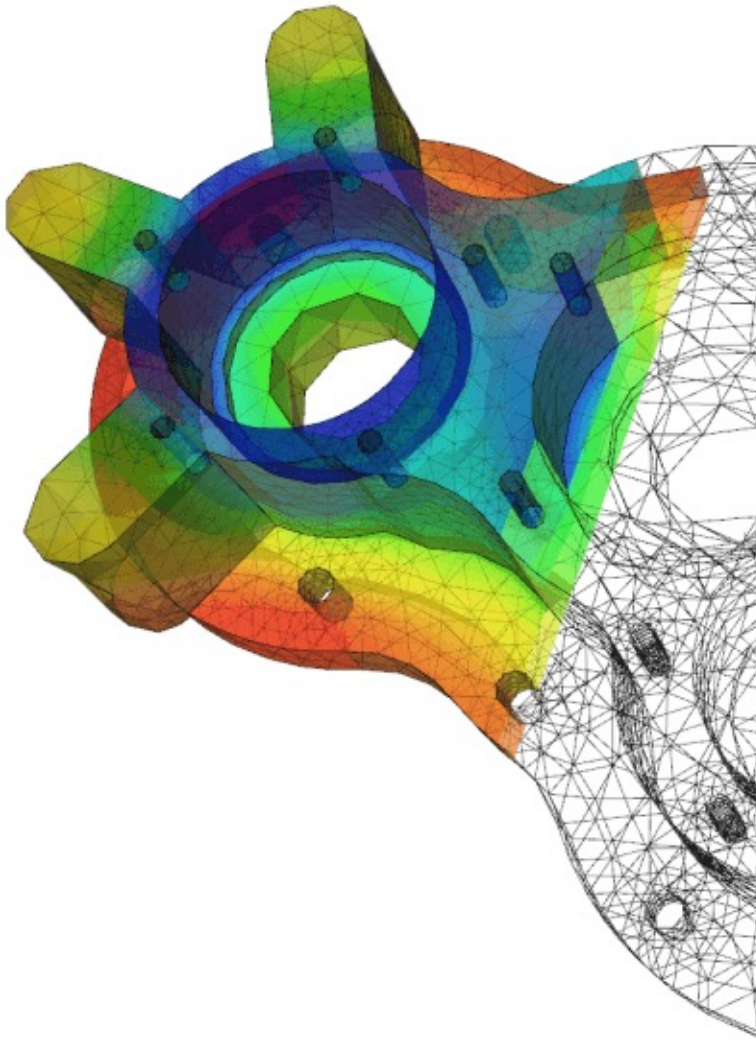
for loop

function call

return statement


Thr106
2.7 Å
3.0 Å
2.7 Å
Met109





Looking back at my PhD, I stumbled into writing a relatively large and complex software project without realising it or being prepared to manage that complexity. […]

I would have welcomed teaching and resources that could have made that process less painful.

Tom Hodson
Physics PhD student, 2021

Problem:

How to support PhD students in their journey from “beginner programmer” to “research programmer”?

Cognitive Apprenticeship model

- programming is a skill
- must be taught in ways that support skill development
- (not just knowledge acquisition!)

Software Engineering

- taking programming skills to a deeper level
- essential for good practice in research programming
- highly transferrable both within and outside academia

Learning by example

- realistic
- applicable
- self-contained
- builds on existing skills
- demonstrates best practice

Research Computing and Data Science Exemplars (ReCoDE)

- based on a real PhD research project
- applicable to a specific domain of research computing
- self-contained codebase
- builds on student's existing programming skills
- demonstrates best practices in software engineering

Exemplar development team:

1x *Graduate Teaching Assistant*

- PhD student with strong programming skills
- domain expert
- proposes exemplar that would be of value to other students

+ 1x *Research Software Engineer* (from central RSE team)

+ 1x *L&T Specialist / Project Coordinator* (from RCDS team)

Learning by working together

- subject-specific knowledge
- best practices in creating replicable, reusable code
- relevant software engineering techniques
- pedagogical aspects – examples and exercises

Languages:

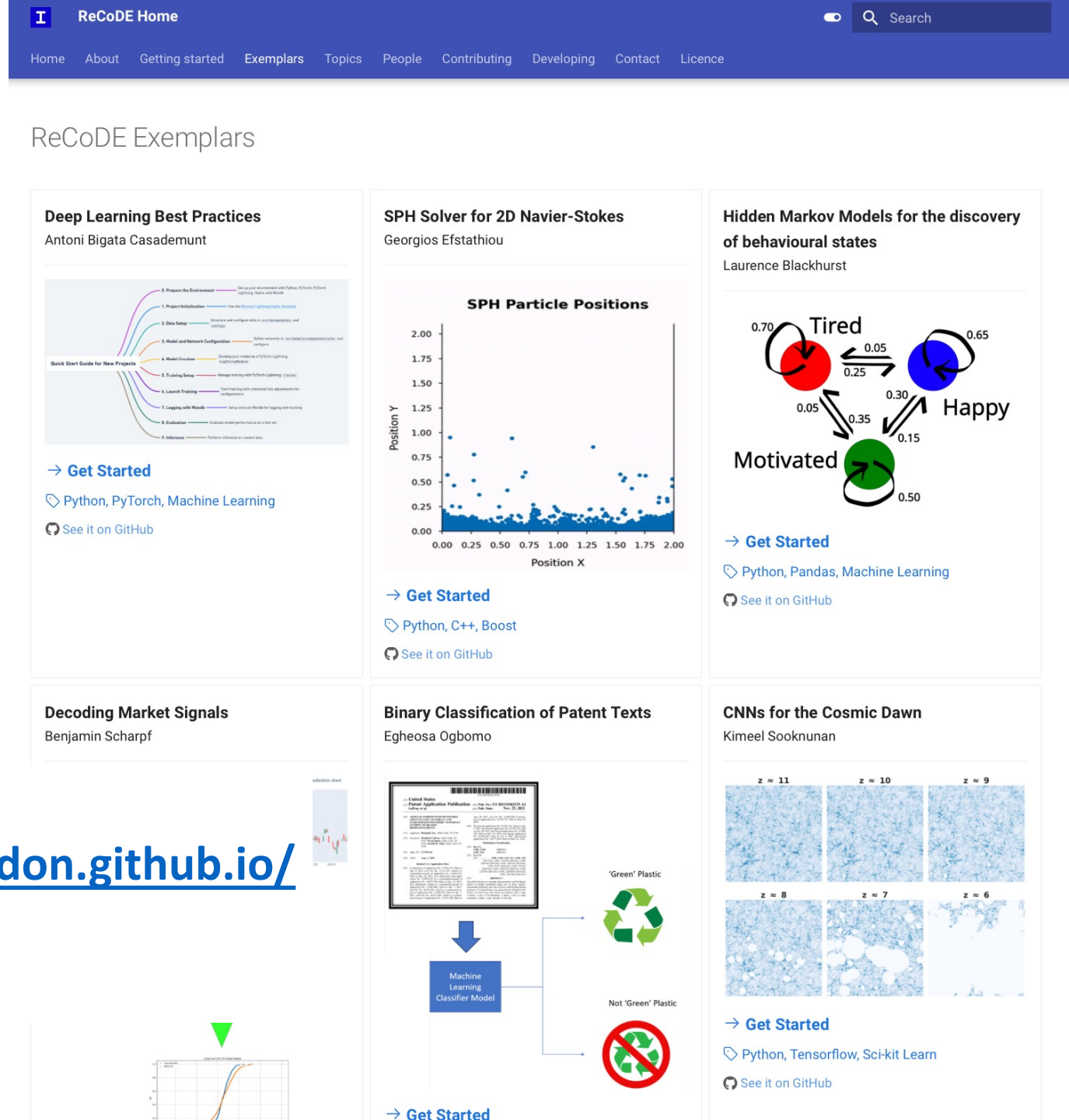
- C++ (1)
- Fortran (1)
- Python (11)
- R (2)

Topics:

Best practices
Boost
CMake
Computer Vision
Convolutional Neural Networks
Data Analysis
Docker
Epidemiology
Finance
GUI
HPC
Logistic Regression
Machine Learning
NLTK
Natural Language Processing
Nextflow
Nuclear Physics
NumPy
Object Oriented Programming
Optimisation
PETSc
Pandas
Patents
Physics
PyTorch
Scikit Learn
Stan
Statistics
Tensorflow
Unit Testing
pyTorch

- 13 exemplars live on website
- 9 more under development
- and further funding sought…

- Student Shapers project to promote ReCoDE over 2024-25



https://imperialcollegelondon.github.io/ReCoDE-home/exemplars

Acknowledgements

Authors

Eliot James Badcock
Laurence Blackhurst
Antoni Bigata Casademunt
Juan Carlos Bilbao-Ludena
Zejian Cui
Bethan C Daniels
Georgios Efstathiou
Fabio Feser
Jack Gisby
Anjali Golding
Tom Hodson
Antonio Malpica-Morales
Emily Muller
Egheosa Ogbomo
Ekin Öztürk
Valentina Quintero Santofimio
Benjamin Scharpf
Kimeel Sooknunan
Jack Trainor
Tycho van der Ouderaa
Shuaixun Wang
Yurong Yu

Student Shapers

Jahnavi Bhaskaran
Giannis Nikiteas

Research Software Engineers

Diego Alonso Alvarez
and the central RSE team

Project Coordinators

Jeremy Cohen
Chris Cooling
Jay DesLauriers
Liam (Jianliang) Gao
Katerina Michalickova
John Pinney

https://imperialcollegelondon.github.io/ReCoDE-home/exemplars