Teaching programming and modelling skills to first-year earth & environmental science undergraduates: outcomes and lessons learned from a pilot project

Jenny A. Fisher, Chris Brewer, Yoke Berry, Glennys O’Brien, Richard Cook
University of Wollongong Australia

jennyf@uow.edu.au  @AtmosJennyF  jennyfisher.github.io/computing-modelling-earthsci/
The Problem

Computing and programming skills are necessary for all scientists.

- large datasets
- 2D/3D visualisations
- numerical models
- complex statistics
- and more…

Traditional geoscience curricula do not teach these skills.
Computing and programming skills are necessary for all scientists.

- large datasets
- 2D/3D visualisations
- numerical models
- complex statistics
- and more…

Traditional geoscience curricula do not teach these skills.

Our solution:
Embed these skills in 1st year Earth & Environmental Sciences core
The Starting Point

Surveys of confidence, interest, & (perceived) usefulness (Hoegh & Moskal, 2009).
The Starting Point

Surveys of confidence, interest, & (perceived) usefulness (Hoegh & Moskal, 2009).

Positive sentiment: 83%  60%  86%

Positive

Negative
Implementation

In-class exercises in Jupyter Notebook:

functions: another example

Let’s look at another example. Recall that the solar flux $S$ in W/m$^2$ can be calculated at a given distance from the sun $r$ in AU using:

$$S(r) = S_0 \left( \frac{r_0}{r} \right)^2$$

where $S_0 = 1386$ W/m$^2$ is the known solar flux at the distance of Earth, $r_0 = 1.0$ AU.

We can define a function called `solar_flux` that takes $r$ as input and returns $S$ as output:

```python
In [5]: # Function to compute the solar flux at a given distance
def solar_flux(r):
    # Define constant variables
    r0 = 1.00 # Distance from sun to Earth in AU
    S0 = 1366 # Solar flux at Earth
    # Use the equation to calculate the solar flux
    S = S0 * (r0/r)**2
    # return solar flux S
    return S

Make sure you understand the syntax of the function. On the first line, we have (r) in brackets - this means that $r$ is an input we are going to provide to the function every time we use it. On the last line we have “return S” - this means that $S$ is the output that we want the function to calculate for us when we give it a value for $r$.```
Implementation

In-class exercises in Jupyter Notebook:

Weeks 1-4: computing fundamentals e.g.:

- simple calculations
- making plots
- using logic (if statements)
- repeating code (for loops)
- writing functions

```
functions: another example

Let’s look at another example. Recall that the solar flux $S$ in W/m$^2$ can be calculated at a given distance from the sun $r$ in AU using:

$$S(r) = S_0 \left( \frac{r}{r_E} \right)^2$$

where $S_0 = 1366$ W/m$^2$ is the known solar flux at the distance of Earth, $r_E = 1.0$ AU.

We can define a function called `solar_flux` that takes $r$ as input and returns $S$ as output:

```python
In [5]: # Function to compute the solar flux at a given distance
def solar_flux(r):
    
    # Define constant variables
    r0 = 1.00  # Distance from sun to Earth in AU
    S0 = 1366  # Solar flux at Earth

    # Use the equation to calculate the solar flux
    S = S0 * (r0/r)**2

    # Return solar flux S
    return S
```

Make sure you understand the syntax of the function. On the first line, we have (r) in brackets - this means that $r$ is an input we are function every time we use it. On the last line we have “return S” - this means that $S$ is the output that we want the function to calculate a value for $r$. ```
Implementation

In-class exercises in Jupyter Notebook:

Weeks 1-4: computing fundamentals

Each exercise followed by Review Quiz for guided, individual practice

instantaneous feedback + hints

point to online resource trinket.io
Implementation

In-class exercises in *Jupyter Notebook*:

- Weeks 1-4: computing fundamentals
  1. Bathtub (basic concept)
  2. Daisyworld
  3. Population
  4. Carbon Cycle
- Weeks 5-8: simple box models

![Flow In and Flow Out diagram](image.png)

![Daisy cover vs. time for Daisyworld](image.png)
Implementation

Weeks 1-4: computing fundamentals
Weeks 5-8: simple box models
Weeks 9-12: creative group projects

e.g.:
How does evolution impact Daisyworld?
How much would a switch to electric cars decrease CO₂?
Could a plague save non-renewable resources?
What happens to food if bees disappear?
How will different emission scenarios change future CO₂?
Impacts on Attitudes

On aggregate: small drop on all measures; only significant change is usefulness
Not all bad news...

More students at high tails of distribution – especially amongst women!
Lessons Learned

1. Students develop proficiency but find process uncomfortable
2. Need incentives to come prepared - pre-lab readings & quizzes help!
3. Need early & frequent positive feedback - alums as student mentors?
4. Need to know how they will use skills in future - share your experience: https://flipgrid.com/858962

All notebooks available online: jennyfisher.github.io/computing-modelling-earthsci/

Thanks to B. Bukosa, J. Greenslade, S. Inakollu, N. Page, & EESC102 students!

Jenny A. Fisher, jennyf@uow.edu.au, @AtmosJennyF
Our Solution

Programming & modelling using Python now central to 1st year core curriculum in University of Wollongong School of Earth & Environmental Sciences

Embedded in practicals for *Earth’s Interconnected Spheres*:
- 1 of 4 compulsory core subjects for all majors
- 100-150 students
- no mathematics prerequisites and no assumed prior programming experience

Why Python?
- free, easy to install, and widely used
- easy to learn “natural” syntax (print “Hello”)
- simple but ultimately powerful, including geoscience integration (e.g. ArcGIS)
Quantitative Measures

Pre- & post-surveys of confidence, interest, & (perceived) usefulness (Hoegh & Moskal, 2009).

e.g.

I am comfortable with learning computing concepts. (C)
I have little self-confidence when it comes to computing courses. (C)
The challenge of solving problems using computing appeals to me. (I)
I would not take additional computing courses if I were given the opportunity. (I)
Developing computing skills will be important to my career goals. (U)
Knowledge of computing skills will not help me secure a good job. (U)

Overall: 8xC, 10xI, 6xU questions map onto three “constructs” (factors)
+ review quiz data
+ assessment marks
+ focus groups / interviews (??)
Gender Differences

*Significant differences in **Confidence, Interest** between male & female students
No significant change except **male students’ perceived usefulness**

Male vs. female **confidence difference remains significant**, interest doesn’t.
More students at high tails of distribution – especially amongst women!
Interest Change

More students at high tails of distribution – especially amongst women!
Future data analysis plans

Waiting on de-identified data to:
• remove students who dropped the subject
• link individual trajectories over time
• link responses to scores (imposter syndrome? Dunning-Kruger?)

Waiting on interviews / focus groups:
• what drives lack of confidence?
• what hurts/helps?