MPS115/116 Presentation Lecture 6: Mini-Project 1

Dr Simon Willerton

We will begin by looking at some generalities of doing a project. Then we will look at the actual project brief.

1 Generalities on doing the project

Overview

You are

- given a mathematical scenario to investigate;
- given a basic task which involves coding with Python;
- required to use that to generate results that you can investigate;
- required to extend the task in some way in which you have freedom;
- required to write this all up as a LaTeX document.

Mathematical investigation

Remember that this is about *mathematical investigation*. You should be using coding, mathematics and your imagination. You should be clearly communicating what you have found out and how.

Before you do anything else

Before starting you should read the brief carefully. Read the brief again. Make sure understand the key points of what is required of you.

Before going near a computer

You should remind yourself of the coding lectures. Sam introduced many important ideas related to writing programs. Do some computations by hand so you have a deeper understanding of what you need to code. This also gives you something to check your code against. Think about what kind of results you expect to get.

Having written some code (some ideas)

- Generate some results.
- Check that you get the same answers as your by-hand computations.
- See how far you can push the code.
- Can you improve the code? Refactor? Use functions?
- Look for patterns.
- Do patterns continue when you generate more results?
- Can you explain or prove the patterns?
- Are there limits to your code?
- Think of other things you might do.

The write-up

- It will be read by your level 1 students.
- It should have a clear structure.
- There should be an informative introduction explaining what is done. (Usually write the introduction last.)
- Don't spend too much time going line-by-line through code. (The markers have access to your code.)
- Do give your results, findings and any conclusions.
- Don't plagiarise.
- Include any references.

Having written a draft

- Reread the brief!
- Make sure you have done what you were supposed to.
- Carefully read through all that you have written.
- Edit.
- Try reading it out loud in your head so that it makes sense.
- Look carefully at the document and see if you can spot any LaTeX problems (as in the previous lecture).
- Given what you have read are there obvious extra things you might try?

Make sure you allow time for these before the deadline!

2 Mini-project brief

Background on the project, the basic task and suggested ways of extending the task are given in the brief available in the assessment section of Blackboard.

The project write-up

You should write up your investigations in a LaTeX report of around three pages, and strictly no more than four, using the preamble template on the course webpage (11pt font, `amsart` document class, `a4paper` option). You should not further alter the layout of the page. The title should be 'Semester 1 Mini-project', and the author should be your registration number. **Do not include your name.**

You should include sufficient discussion of your Python code as to explain how it works, but a line-by-line description of the code is not necessary. The code should be written with meaningful variable names and good use of comments to aid readability. You should also include results of your investigations with some discussion and analysis. Appropriate references must be given for any sources used.

Your project will be *peer-assessed*. This means that it will be marked anonymously by your fellow students. You will receive a mark out of 8 for the project itself, and a mark out of 2 for your involvement in the peer-assessment. Overall, the project counts as 10% of your final module score.

You will submit PDF, LaTeX and Python files online via a dedicated website. The deadline for uploading the projects is 2.00pm on Wednesday 20th November (Week 8). More details will follow. [2em]

Note that the peer assessment session will be in the standard MPS115/116 lecture slot on Tuesday 2nd December (Week 10). Please make sure that you attend. This will be in a different room to normal.

Late work and academic misconduct

Late work

Due to the peer-assessment of this work, it is important that work is submitted on time. Any work submitted after the deadline may be given a mark of zero. If you have extenuating circumstances affecting your ability to hand in the work then you must submit the Google form in plenty of time.

Academic misconduct such as plagiarism

It is essential that you are familiar with the lecture that was given on academic misconduct as ignorance will not be a defence. The mini-project is an individual assignment and must be your own work. You must not copy work from other students and must not allow other students to copy your work. You can ask for help on the module discussion board, but must not

ask for help on any other internet discussion forum or email list or anything of a similar nature. Use of AI tools, for example ChatGPT, is forbidden. All sources must be cited. If academic misconduct is suspected then we will follow university procedures and penalties will be imposed where appropriate, e.g., a mark of zero.

Advice for a good project

Here is some advice for how to write a good project.

- Use the module discussion board. We may post hints on there.
- Re-read the Python lecture notes on how to write a program and how to improve the chances of finding bugs easily.
- After making a start on the project, carefully re-read this brief, as well as the marking descriptors, to make sure you have understood what is expected of you.
- Make your Python code as tidy as possible, with comments where appropriate; after all, it will be read by your fellow students.
- Make your write-up as readable as possible by following the advice in the presentation lectures and learning from your homework feedback.

We hope this mini-project will help you to consolidate what you have learnt so far and that you enjoy doing it. Please direct any questions to the discussion forum.

3 The computer lab

The computer lab

In the lab you will see how to display Python code in LaTeX.