Problems 2023

Programming And Scripting

Due: I will probably correct these in April, so try and finish before that. (You should be working on the project then)

This document contains the instructions for Problems (weekly tasks) for Programming and Scripting.

Every week a new problem will be put into the weekly tasks on Learnonline. Your solutions will be assessed towards the end of the semester and will be worth 50% of your marks for this module. The marking scheme is given below. It is expected that you will be working on the exercises throughout the semester. It is not expected that you get every program right first time. So long as an attempt is made the week the problem is posted, this will count as a good approach. It is important that you keep working on any incomplete problems until the deadline. Please note that all students are bound by the Quality Framework [3] at GMIT which includes the Code of Student Conduct and the Policy on Plagiarism. The onus is on the student to ensure they do not, even inadvertently, break the rules. A clean and comprehensive git [1] history (see below) is the best way to demonstrate that your submission is your own work. It is, however, expected that you draw on works that are not your own and you should systematically reference those works to enhance your submission.

How to submit

Your solutions should be pushed up to a repository called ***pands-problem-sheet*** on your GitHub account (please discuss with me if you wish to use another provider). This repository should only contain files that relate to the problems set in the weekly tasks. You should paste the URL for this repository in the assignment created on Learnonline in the problem sheet section. (You may continue to push changes to the repository until the due date). You should be able to resubmit this URL if you need to change it.

The Marking Scheme

This problem set will be worth 50% of your marks for this module. The following marking scheme will be used to mark your submission out of 100. The examiner's overall impression of the assignment may influence marks in each individual component.

Area	Description	mark
Development	Clear, well-written, and efficient code.	30%
ReadMe File	Concise, Clear readme file	20%
Comments and file names	Comments in Code file, indicating you understand what is happing. Using the correct file names. (This mark may be combined with the readme file, or you can choose to use Jupyter notebooks)	15%
Research	Evidence that you have researched solutions to the problems (References)	20%
Consistency	Regular commits with clear commit messages. There will be marks given for each week that there is a commit with a relevant comment	15%

Advice for students

- You must be able to explain your assignment during and after its completion. If you had trouble understanding something in the first place, you will likely have trouble explaining it a couple of weeks later. Write a short explanation of it into your submission, so that you can jog your memory later.

- Everyone is susceptible to procrastination and disorganisation. You are expected to be aware of this and take reasonable measures to avoid them.

- Students have problems from time to time. Some of these are unavoidable, such as acute family issues or illness. In such cases allowances regarding assignments can sometimes be made. Students should be able to show that up until an issue arose, they had completed a reasonable and proportionate amount of work and took reasonable steps to avoid preventable issues.

- Go easy on yourself — this is one assignment in one module. It will not define you or your life. A higher overall course mark should not be determined by a single assignment.

FAQ:

Is this repository different than the one that was submitted in Week01?

Yes, when you are finished this course you will have three repositories in your GitHub account

1. ***my-work***: The one you submitted in week one where you push your own work, I may use this if there in extenuating circumstances to evaluate how you were getting on with the course. But officially there are no marks being awarded for code you push to this repository.
2. ***pands-problem-sheet:*** The repository for the problem sheets, this should only contain the solutions to the problems and supporting documentation (eg readme file).
3. ***pands-project:*** I will be setting a project for you to do later in the course, this repository is where you will push that.

Very few marks are being given for the code, Why?

Other modules will concentrate more on the code you submit, I have taken a conscious decision to focus on looking at your process of writing code, so I am giving more marks than other modules for aspects like comments, consistency of commits and references. Be aware that other modules in this course may focus more on the coding aspect.

Should I be able to complete each problem on the week it was submitted?

No, I am deliberately setting problems that you will have to search for an answer from somewhere other than my course material. I do expect you to push your first attempt to each problem on the week it was released, but you may improve upon each submission up until the deadline.

How can I make my readme file look nice? (you may ask this much later in the module)

Readme files use a format called markup, I leave the rest to you. (You can also use Jupyter notebooks).

References

[1] Software Freedom Conservancy. Git. https://git-scm.com/.

[2] Inc. GitHub. Github. https://github.com/.

[3] GMIT. Quality assurance framework. https://www.gmit.ie/general/quality-assurance-framework.