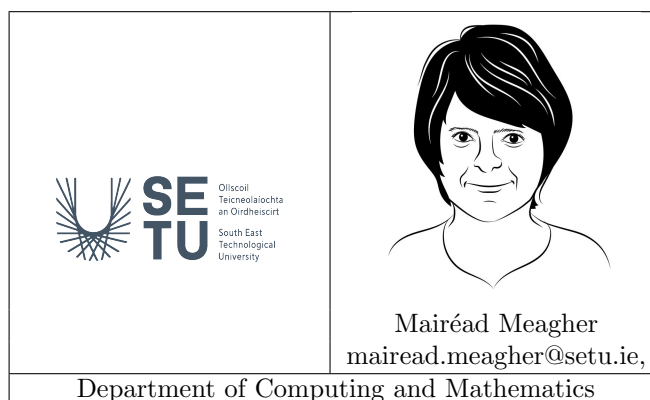


Functional Programming Course Outline and Essential Information

Semester 2 - '22-'23

Mairéad Meagher, SETU.



Contents

1	Module Name	3
2	Lecturer	3
3	How to reach me	3
4	Learning Approaches	5
4.1	Learning Technologies	5
4.2	Structure of lectures/ labs	6
5	Module objectives / Learning outcomes	7
6	Assessment Breakdown	8
7	Academic Integrity	9
8	Important note about engagement in the module and time management	11
	Course Outline and Essential Information- Functional Programming	1

9	Netiquette and Decorum	12
9.1	Netiquette	12

1 Module Name

Functional Programming

This is the year 4, semester 2 running of this module in South East Technological University.

2 Lecturer

Mairéad Meagher,

Lecturer in Department of Computing and Mathematics,
School of Science and Computing, South East Technological University.

3 How to reach me

- The quickest way to reach me is via Slack (please join the Functional Programming Slack workspace here).

I will be using this as the main form of communication for this running of the module.

- You can also reach me via email: mairead.meagher@setu.ie.
- I am available during work hours from Monday to Friday, 9am to 5pm. You can email me / Slack me outside of these hours and I will reply as soon as I can, but always within three days.(If this does not happen, assume your contact has gone into spam etc. and please re-contact me.) When emailing me, please indicate what module you are taking as well as the nature of your query in the subject line, and do not forget to use an appropriate greeting and sign off.

It's important to be polite and to treat one another with respect; let's start as we mean to go on.

4 Learning Approaches

4.1 Learning Technologies

We will be using

- **Moodle** - our learning management system, where you can find links to the notes. If you are not registered here, let me know asap. Each week you will be given a todo list of materials to have covered/labs exercises to have completed before the next week.
- **tutors** - This static website will hold all the notes, labs and links to videos. There will be a pre-recorded lecture associated with each lecture. See Figure 1. This site has the course published as far as we have covered at any time so you can navigate in whatever way suits you.

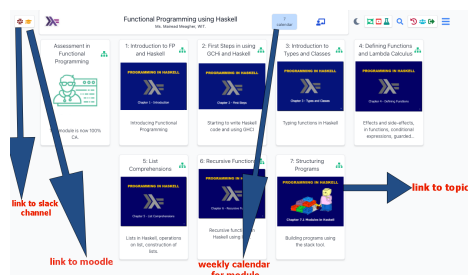


Figure 1: Example of tutors website

4.2 Structure of lectures/ labs

Your timetable structure is :

- 14 weeks (12 weeks of tuition)
- 4 hours per week.

During each of these contact hours, we will have (inter-changeably) either/both:

- Lectures
- Labs practising techniques we have seen in lectures
- Working through/reviewing exercises sheets.

Because of this fluid structure, **I will ask that you bring your laptops to all classes.** If any of you have a problem with this, contact me as a matter of urgency.

The module is broken down into 12-14 topics. These topics are curated on tutors. Each topic will have associated with it:

- Lecture/s (there will be a link to a pre-recorded lecture with each slide deck)
- Labs - either practical work using Haskell or exercises sheets.

5 Module objectives / Learning outcomes

On completion of this module students should be able to:

1. Construct simple and more complex programs in Haskell;
2. Construct basic constructs of Haskell;
3. Use tools to help build Haskell projects, notably GHCi and Stack;
4. Improve the scope of programs by using Haskell libraries;
5. Write programs to solve problems in specific domains, e.g. parsing;
6. Write Haskell programs that call other languages and vica versa.

GHCI lab test	lab test Stack	Prog Assign 1	Prog Assign 2	Final Exam
5%	5%	20%	20%	50%
Week 4	Week 7	Week 10	Week 13	Exam Session
11:15	11:15	18:00	18:00	tbc
Thur 9th Feb	Thur 2nd March	Sunday 2nd April	5th May	(exams office)
tbc	tbc	tbc	tbc	tbc
		Interviews: Week 11	Interviews Week 13/14	

Table 1: Assignment Schedule

6 Assessment Breakdown

The assessment in this module is made up of continuous assessment during the module and a Final Exam at the end of the semester. The schedule is seen at Table 1 - Assignment Schedule.

1. Continuous Assessment (50%) broken into

- In class lab test - 5% - starting using GHCI – this is to ensure that all students are competent in using GHCI and writing small Haskell programs. It includes fixing syntax errors, e.g. filling in missing type declarations in simple functions.
- In class lab test - 5% - using Stack – a simple program using the Stack structure.
- 20% programming assignment 1
- 20% programming assignment 2

2. Written Final Exam (50%) This will be a written final exam.

The CA is rolled out in the above order so that you, the student, can steadily build up marks throughout the semester.

For all the continuous assessment submitted during the module, you will get your marks back as soon as is possible, but usually within a week. If you are wondering why you got a particular mark, **always** ask me. My marking schemes are very comprehensive and I am happy to go through the breakdowns with you. I don't give this comprehensive feedback by default to speed up the return of the marks but am happy to engage about them later.

For the in-class lab tests, I will give you sample lab tests so that you know the nature of the tests beforehand.

For the programming assignments, a marking scheme will be published with the specification of the assignment. As always, be sure that you are aware of the marking scheme. If there are marks going for a particular part, and you haven't attempted that part, there is nothing I can do. Always make it easy for the examiner to give you marks!

If you wish to seek an extension for an assignment, you must do so in sufficient time (i.e. not on the day of submission, and not when the submission date has passed) and must provide a valid reason for seeking the extension.

7 Academic Integrity

The School of Science and Computing at South East Technological University are committed to maintaining the highest standards of academic integrity. Academic misconduct, including, but not limited to, cheating may result in a mark of zero for the assignment as well as dis-

ciplinary action. Additional sanctions may be imposed depending on the case. You are responsible for ensuring that you do not get involved in cheating of any kind.

With regard to programming submissions, an interview is mandatory and is part of your assignment mark (as a multiplier). The interview is to ascertain that the work is your own and that you fully understand how it works, in its elemental parts and how it works together.

We will always encourage you to work in collaboration with your fellow classmates. But please be careful not to cross the line between collaboration and using someone else's work. Please do not be tempted to use this route. It is too risky and the penalty can affect your academic future.

8 Important note about engagement in the module and time management

Part of active engagement for any module involves a degree of time management. As part of this module I will be asking you to complete exercises, between class times. In some cases I will be asking you to read material before class. (These tasks will be clearly signalled to you at the time). These will not be graded but, by engaging in these tasks at the time, you will be in a better position to understand the next part of the module. We will approach the module in a step-by-step manner, so opting out at any part will make it more difficult for you to keep up. This is where time management will come in - you need to be careful to ensure that you keep a balance between modules.

Always ask questions, either in class or in a breakout room. One way to help to stay engaged is to ask questions if you don't understand what is going on. Remember, when you are asking questions:

1. Just the process of asking a question means that you have learned something.
2. If you cannot understand, in most cases, you are not the only one.
3. Asking questions means that the pace of the lecture/labs will suit you better - we will always keep going if there are no questions!

9 Netiquette and Decorum

In all of our asynchronous discussions online, mostly Slack for us, it is important that we foster a supportive, safe, and engaging learning environment where we can critically discuss, analyse, and reflect on the readings and topics each week. Diverse views are encouraged and welcomed and should be based on evidence. You are free to express your views and ideas as long as your words or action do not demean, intimidate, or intend to violate the rights and dignities of others. Hate speech is not acceptable and may result in disciplinary action. Hate speech includes words or actions that threaten or target the safety and liberties of an individual or group.

9.1 Netiquette

The word netiquette is a combination of 'net' (from internet) and 'etiquette'. It means respecting other users' views and displaying courtesy when posting your views to online discussion groups (see BBC)

- Remember that there is a human being on the other end of your communication
- Treat that human being with respect
- Do not post a message that you would not be willing to communicate in a face to face environment.
- Keep it courteous
- Be kind and professional: Online communication comes with a level of anonymity that doesn't exist when

you're talking to someone face-to-face. Sometimes this leads people to behave rudely when they disagree with one another. Online students probably don't have the complete anonymity that comes with using a screen name, but you could still fall prey to treating someone poorly because of the distance between screens. Make a point to be kind and respectful in your comments—even if you disagree with someone.

- Extend your good nature online: The digital world is an increasingly important part of our lives. We should be our best selves there too. The manners our parents taught us apply everywhere.
- Promote healthy discussions: To get the most out of online forums, a useful netiquette guideline is to promote healthy discussion. You can help your online community by posing questions, sharing experiences, providing positive feedback, asking follow-up questions, and referring to information sources. Being a positive contributor is better than being a critic, troll or other negative force.
- Ignore inflammatory comments by trolls It's generally best to ignore trolls. These are internet users who try to bait other users into a reaction. Trolls might be honest in what they're saying, or they could be sarcastic or deliberately dishonest. You can tell a troll by the inflammatory nature of their statements. They want to stir up negative emotions and responses.

- Respect others as equals: Show a little respect and humility online. Think – that 'idiot' who wrote the opinion you completely disagree with is a human being. They have feelings and experiences. They may believe passionately in what they're saying. And they may actually be right. Even if you're feeling dismissive or knowledgeable or whatever, inject respect into your writing. That's just being fair to others.
- You're here to learn and contribute, not dictate: While we all like to think that our opinion matters, you'll gain more from internet forums by approaching them as a learner. When everyone is trying to express their view rather than hearing from others, forums become noisy, crowded with posts, and disjointed. A more polite and effective path is to adopt a listening mode. Read posts carefully, ask questions, and write something only if it offers value to the discussion.
- Read first Take some time to read through each of the previous discussion post responses before writing your own response. Remember, discussions can move fairly quickly so it's important to absorb all of the information before crafting your reply. Building upon a classmate's thought or attempting to add something new to the conversation will show your instructor you've been paying attention.
- Remember, your words are permanent: Be careful with what you post online. Once it's out there, you may not be able to get it back.

- Make your point in a nice way: Write in a way to get the kind of reaction you want. A little thoughtfulness, strategy and netiquette can go a long way in online discussions. Your first draft of an online post is unlikely to be your best. Are you disagreeing with someone in a flippant way? Have you misinterpreted what they really meant? Will you put people off with the tone of your text.
- Pause before you post: It's worth taking a moment to reflect before hitting the send button. When you're using a computer, you're normally clicking, and scrolling and typing all over the place. Most things are done quickly. But one time when it's important to slow down is when you're about to post something online for all the world to see. Pause and reflect for a second. Are you truly comfortable with what you're sending?
- Respect the opinion of your classmates: If you feel the need to disagree, do so respectfully and acknowledge the valid points in your classmate's argument. If you reply to a question from a classmate, make sure your answer is accurate!
- Forgive and Forget If you're offended by something another student says online, keep in mind that you may have misunderstood their intentions. Give them the benefit of the doubt.