Programming Fundamentals 1
Semester 1 - 2025-2026

Course Outline and Essential Information

Module Lecturers

| Dr Siobhán Roche, | Mairéad Meagher |
| siobhan.roche@setu.ie | mairead.meagher@setu.ie |

Department of Computing and Mathematics

Ollscoil
Teicneolaíochta
an Oirdheiscirt

South East
Technological
University

Contents

1 Module Name

Programming Fundamentals 1

2 Lecturers

Siobhán Roche, siobhan.roche@setu.ie, Lecturer in Department of Computing and Mathematics.
Siobhàn will take lectures and labs for groups W3 and W4 in this module.
Mairéad Meagher, mairead.meagher@setu.ie, Lecturer in Department of Computing and Mathematics.
Mairéad will take lectures and labs for groups W1 and W2 in this module.

3 How to reach us

- The quickest way to reach us is via Slack. Please join the Programming Fundamentals 1 Slack workspace here. We will be using this a main form of 'outside class' communication for this module. If you have a general query, please use the **#general** channel. If you need to contact us directly, please DM one or both of us rather than using public channels. You should normally contact the lecturer that you see face-to-face about most matters (i.e. either Mairéad or Siobhán) but feel free to copy us both.

- You can also reach us via email:

  - siobhan.roche@setu.ie and\or
  - mairead.meagher@setu.ie.

  We are available during work hours from Monday to Friday, 9am to 5pm.

  You can email/Slack us outside of these hours and we will reply as soon as we can, but always within three days.(If this does not happen, assume your contact has gone into spam etc. and please re-contact us.) When emailing us, 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 Technologies

- All lectures and labs will happen in person as per timetable.

- We use tutors to publish the material for this module. This static website will hold all the notes, labs and links to videos. See Figure 1. This site is organised by topic,as opposed to Moodle which is organised by week(by linking to, among other things, the relevant tutors topic).
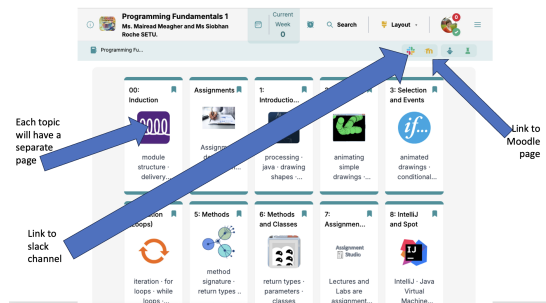

Figure 1: Example of tutors website

There are a number of visual themes available across the site. See Figure 2. Of particular note, there is a font which is dyslexia-friendly. This is marked on the diagram.
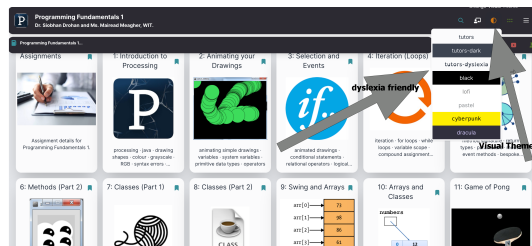
Figure 2: Visual themes in tutors

- **Moodle** - our learning management system(used all over SETU, Waterford campus), where you can find links to the notes, and where online exams are curated. The link to this course is here. Each week you will be given a list of activities to have done before the class and a list of activities to have done before the next week. The link to the Moodle page for this module is here.
  Each week there will be a set of links made available to you for that week including:

put in Moodle link

insert Moodle link

- Links to topics in the tutors website (see below)

- Links to 'dropboxes' for the labs for that week (see below)

- **Slack** - our main form of communication outside of class. See above for links.

- **GitHub** - we will be using GitHub for the tutors website. **To use tutors, you will need to have a GitHub account. (https://github.com/) For the moment this is not necessary but we will turn this system on in Week 3.**

5 Module objectives / Learning outcomes and schedule of teaching

5.1 Module objectives and Learning Outcomes

On completion of this module students should be able to:

1. Apply core problem solving approaches suitable to the programming discipline to build algorithms.

2. Write small applications using standard sequence, conditional and iterative control structures.

3. Modify and expand small applications.

4. Write small applications that use simple UI, computation and data structures.

5. Develop techniques to effectively test, debug and document small applications.

6. Analyse and explain how the above applications work.

7. Write a simple object-oriented program using the *BlueJ* environment.

   Note: We do not expect you to have **any** prior experience of programming, either in Java or any other language. We will start from the very beginning and build up from there.

   For this module, we will be using Java, and using the ***blueJ***[1] environment.

   Note that note that ***blueJ*** is available free to use. Details of how to download it will be in the labs before we use them.

---

[1]https://www.bluej.org/

5.2   Schedule of Teaching

The schedule of teaching is given in Table 3. Week 1 will start on Monday 15th September. This may change slightly as the semester progresses.

Semester 1		⊗
Week No.	**Type**	**Date Starts**
0	Development of the course	Jun 1
1	Topic 01 - Introduction to the Module, Java and BlueJ	Sep 8
2	Topic 02 - Variables and Data Types	Sep 15
3	Topic 03 - Inputs and Expressions	Sep 22
4	Topic 04 - Selection	Sep 29
5	Topic 05 - Loops	Oct 6
6	Topic 06 - Arrays	Oct 13
7	Topic 07 - Methods	Oct 20
Reading Week	Reading	Oct 27
8	In class exam (40%) & Topic 08 - Testing	Nov 3
9	Topic 09 - Introduction to Classes and Objects	Nov 10
10	Topic 10 - Encapsulation and Interaction	Nov 1
10	Topic 11 - Arrays of Objects	Dec 2
11	Assignment Studio - Assignment Studio and Handup	Dec 2
Study Week	In class test and Interviews	Dec 9

Figure 3: Week-by-week schedule of teaching

This list of weekly topics is also available on the tutors page for this module - top row.

Assessment	In-class Test	Programming Assignment	Tutorial attendance
Week No.	Week 8	Week 12	each Week
Time of test/ Time /Deadline	During class time	18:00, Sunday 7th Dec., 2025	n/a
Written exam	none	Monday 8th Dec	n/a
Interview	n/a	Week 12/13	n/a
Percentage	40%	50%	10%

Table 1: Assignment Schedule

6 Assessment Breakdown

The assessment in this module is **100%** Continuous Assessment made up of three components. There will be no final examination in this module.

6.1 Continuous Assessment (100%)

Your module assessment is made up of:

- **Attendance at tutorials (10) - 10%.** You are asked to attend your tutorials where written problem sheets will be given and so that you can practice any new concepts.

- **Written in class test - 40%.** This will be a test based on the material covered in the first half of the module and largely based on tutorial sheets and labs.

- **Programming Assignment - 50%.** Having practiced the material in the first half of the module, you will be asked to complete a programming assignment which uses the material covered in the first half of the module and the new material covered in the second half of the module.

  You will be asked to solve a problem using classes, methods and arrays. The problem will be fully specified and you will be asked to implement it in **Java** using the *blueJ* environment

In the case of the in-class test, 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 us. Our marking schemes are very comprehensive and we are happy to go through the breakdowns with you. We don't give this comprehensive feedback by default to speed up the return of the marks, but are happy to engage with you about them later.

In the case of the programming assignment, the marks are not published as the final mark is overseen by the external examiner and the finalised mark is released only after the examinations are fully processed. to sit an in-class test based around your assignment. This is to help us during our interview process

for this assignment. However, we will be happy to go through those marks with you after the results have been processed and finalised.

Also, in the case of this programming assignment, you will be asked to sit an in-class test based around your assignment. This is to help us during our interview process for this assignment. Please take this opportunity to clarify any queries you have. This is a good way to ensure that you are attempting the work in a suitable manner.

For the programming assignments, a **marking scheme will be published with the specification of the assignment**. 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, we cannot give you any marks for that part! .

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 Eastern Technological University (Waterford) 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 disciplinary action. Additional sanctions may by imposed depending on the case.

You are responsible for ensuring that you do not get involved in cheating of any kind. This includes non-referenced code, AI generated code, copying code from the internet, copying code from another student, allowing another student to copy your code, etc..

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 Engagement in the module and Time Management

8.1 Engagement

Part of active engagement for any module involves a degree of time management. As part of this module we will be asking you to complete exercises, between class times. Each week, for instance you will be asked to complete labs (available on Moodle) and upload your attempt on Moodle. 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 and to attempt the assignments. 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 during labs. 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!

8.2 Time Management

There are many software tools available to help you managing yout time and in general helping you to plan your work. Have a look around - see if any of them suit you. You should **not** have to pay for one there are plenty Open Source versions and in some cases there are free educational versions (always check this out) A couple that are worth mentioning:

- **Trello** - this is great Project Management Tool - it's Open Source and used in the software industry

- **Notion** is a commercial product but free to students - you just need to register. There are many templates available for your use - it's well worth a look.

- A notebook!

Learning how to organise your material/ thoughts / ideas is a very good use of your time. It may be one of the most valuable things you learn.

9 Organizing yourself

9.1 Organisation of your labtop/cloud storage

It is really important to organise where you store all your college files/ software. If you don't do this early, chaos will ensue! The best way is to organise your files from the start. You can use this file structure on Cloud (One-Drive?) and mirror it on your labtop so you can 'backup' easily by dragging the full folder from one to the other. If you are using a labtop, then you can use your local drive but be sure to back it up regularly.

9.2 Suggested Filing System

A suggested filing system is given in Fig 4. Note you could have subfolders named Week 1 etc. whichever is most appropriate.

Having set this up, you will get used to using this very quickly and you will save a lot of time in the long run.
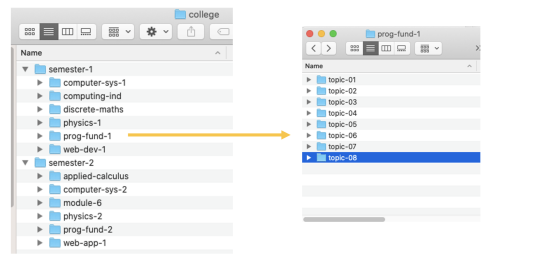

Figure 4: Suggested Filing System

DO's	DON'T's
Set up the folder structure and continue to use it.	Set up and use it 'an odd time'.
Set up 'favourites' folder in Explorer/Finder for your 'college' subfolder	Use Downloads or Desktop as the root folder for 'college' subfolder.
Store data (weekly homework, etc.) using this structure.	Mix up data and programs in college folder.
Store your software (e.g. BlueJ) in another folder e.g. /dev or Program Files.	

Table 2: Do's and Don't's of Filing Systems
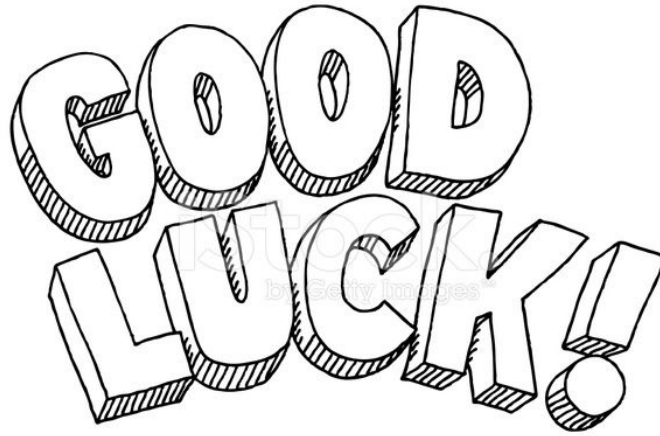
Finally, we wish you all the best of luck with this module and all of your university experience,

Siobhán and Mairéad.