

INF231:
Functional Algorithmic and Programming
Lecture 0: General Information

Academic Year 2020 - 2021

$f(x)$



Some practical information

Lecture sessions:

- ▶ Romain Couillet
- ▶ Wednesday, 9:45am - 11:15am
- ▶ *one exception*: 2nd class on first week: Friday, 9:45am - 11:15am

Exercise sessions (TD):

- ▶ Romain Couillet
- ▶ Tuesday, 3:15pm - 4:45pm

Practice sessions (TP):

- ▶ Romain Couillet
- ▶ Thursday, 3:15pm - 4:45pm (+ 4:45pm - 6:30pm if needed)

Caseine: For course interaction, contact me on [Caseine](#)

Email: romain.couillet@gipsa-lab.grenoble-inp.fr

Some practical information

continued

Web pages:

- ▶ your main link: <https://moodle.caseine.org/course/view?id=434>
- ▶ my personal webpage: <https://romaincouillet.hebfree.org/index.html>

Phone numbers:

- ▶ +33 4 76 82 64 53

Office location:

- ▶ GIPSA-lab, 11 rue des Mathématiques, Building B, Office B373
(in front of tram stop Gabriel Fauré)

Meetings are possible (on appointment).

One week of INF 231 is:

- ▶ 1 Lecture session (cours): 1h30 (so far in visio)
- ▶ 1 Exercise session (TD): 1h30 (so far in visio)
- ▶ 1 Practice session (TP): 1h30 (so far in visio)
- ▶ A lot of Personal Work:
 - ▶ 1 practice session on your own
(finishing practical assignments, project, ...)
 - ▶ Autonomous work

Teaching Material

- ▶ Lecture Notes (slide handouts)
- ▶ Website (mainly Caseine)
- ▶ OCaml interpreter: to install first of all on your laptops!
<https://moodle.caseine.org/mod/book/view.php?id=35853>
- ▶ Also available is an online version (to use with care!)
<https://try.ocamlpro.com/>
- ▶ References

Getting ready to start!

Everything is on Caseine:

<https://moodle.caseine.org/course/view.php?id=434>

- ▶ **subscribe to exercise session (TD) groups!**
- ▶ TP = groups of 3 students (*automated*)
- ▶ locate the links to classes, material, lab report dropout, etc.
- ▶ explore Caseine!

Assessment

continuous

- ▶ Mid-term exam (during week 6 or 7)
- ▶ All your lab reports
- ▶ Project
- ▶ Final Exam

Final Grade = 60%.Final Exam + 20%.CC1 + 20%.CC2

CC1= 40%.TP + 60%.Project (to be confirmed)

CC2= Midterm Exam

This can change according to sanitary conditions and rules!

Some Advice

Sounds naive but they are the key to your success

Pay attention in the lectures: *Never get out of a lecture room without having an understanding of everything. Exercise sessions are not purposed to understand the lectures but to practice.*

Ask questions: *If you have a question, at least two of your fellows have the same question.*

Work hard and on a regular basis: *Thinking that you can assimilate the content one week before the exam is illusory.*

Write things down! *Coding and algorithms can get very abstract: writing on paper helps a lot.*

Don't hesitate to contact me if you are lost on something:

- ▶ *I am available and willing to help.*
- ▶ *Try to solve the problem by yourself→ try with your fellows→ send me an email.*

Don't get lost in the middle of the semester / Never give up!

*"I hear, I forget,
I see, I remember,
I do, I understand"*
Confucius

References

- ▶ Guy Cousineau et Michel Mauny. *Approche fonctionnelle de la programmation*. Ediscience (Collection Informatique), Paris, 1995, ISBN 2-84074-114-8.
- ▶ Emmanuel Chailloux, Pascal Manoury et Bruno Pagano. *Développement d'applications avec Objective Caml*. Editions O'Reilly, Paris, 2000, ISBN 2-84177-121-0.
- ▶ Xavier Leroy et Pierre Weis. *Manuel de référence du langage Caml*. InterEditions, Paris, 1993, ISBN 2-7296-0492-8. Version électronique
- ▶ Ocaml Inria web site
- ▶ Ocaml Reference Manual
- ▶ Ocaml Interpreter (online or not)
- ▶ Programming Conventions in Ocaml:
 - ▶ <http://caml.inria.fr/resources/doc/guides/guidelines.fr.html>
 - ▶ http://www.seas.upenn.edu/~cis500/cis500-f06/resources/programming_style.html

Lecture Agenda

4 main parts

- ▶ Types, expressions, functions
- ▶ Recursion
- ▶ Higher-order (functions)
- ▶ Tree-based Structures

Acknowledgments

This course has been previously taught by Michaël Périn, Francois Puitg, and Thanh Hai To.

Lecture slides are based on Thanh Hai To's latest class, themselves based on:

- ▶ some previous lectures by Michaël Périn, François Puitg, and Thomas Braibant
- ▶ lecture notes of Jason Hickey - *Introduction to OCaml*