CS 3100: Paradigms of Programming

Jul-Nov Semester 2021, E-Slot (3 Cr)

Instructor : **Jayalal Sarma**

Lectures : Tue @11am, Wed @10am, Thu@8am at the Webex Link

jayalal@cse.itm.ac.in, 9840920902

Instructor Contact Hours : Tue, 3:30pm - 4:30pm at the same webex link

TA contact hours : Thu, 10am - 11am.

Updated on August 16, 2021

Note: Course related communications will be on the WhatsApp group since all students have been successfully added to the group. Important announcements will be sent to moodle forums as well.

1 Course objectives

The objective is to learn various prevalent programming paradigms, and to compare them in the context of various algorithmic implementations. The students would learn the fundamentals of each paradigm and how a difference in the paradigms affects programs. Outcomes: 1. Choose an appropriate paradigm for a given problem. 2. Solve a problem in multiple paradigms. 3. Learn nuances of functional and logic style programming.

Since the students are already exposed in a very elaborate way to imperative programming (and in particular object oriented programming), substantially more time will be allotted to functional and logic programming in the course.

2 Course Syllabus

Introduction: to different paradigms of programming: Role of Types - Static and Dynamic Type Checking - Scope rules ; Grouping Data and operations, Information Hiding and Abstract Data Types, Objects, Inheritance, Polymorphism,Templates.

Functional Programming: Expressions and Lists, Evaluation, types, type systems, values and operations, function declarations, lexical scope, lists and programming with lists, polymorphic functions, higher order and Curried functions

Logic Programming: - Review of predicate logic, clausal-form logic, logic as a programming language, Unification algorithm, Abstract interpreter for logic programs, Semantics of logic programs, Programming in Prolog.

Languages used will be **OCaml** and **Prolog** for the second and third module respectively.

Note that the first module is not being planned to be taught separately. Since the students have been exposed in detail to object oriented programming paradigm, we will be ensuring that most of the topics comes in naturally in the third and fourth modules and will be discussed in those sections.

3 Reference Textbooks

- Functional Programming in OCaml - Sep 2020 edition. A textbook based on courses taught by Michael R. Clarkson, Robert L. Constable, Nate Foster, Michael D. George, Dan Grossman, Daniel P. Huttenlocher, Dexter Kozen, Greg Morrisett, Andrew C. Myers, Radu Rugina, and Ramin Zabih.

- Programming Languages: Concepts and Constructs; 2nd Edition, Ravi Sethi, PearsonEducation Asia, 1996.

- Programming Languages: Design and Implementation (4th Edition), by Terrence W. Pratt, Marvin V. Zelkowitz, Pearson, 2000.

- Programming Language Pragmatics, Third Edition, by Michael L. Scott, Morgan Kaufmann, 2000.

4 Lectures and Mode of Teaching

There are three lectures a week. This makes it total of 42 lectures in the whole semester. Each lecture is planned as a live lecture on webex by using Jupiter Notebooks on shared screen. The notebook is later distributed through

github repository. The recorded videos of the live lecture are made available through shared google drive folder.

5 Evaluation

The exams will be conducted online on moodle with recording of entire screen, camera visuals. The students must ensure that they have network connection required for the same on the day specified. Of course, unexpected events like power outages, will be handled separately. If there are genuine difficulties to arrange these, the respective students are requested contact the instructor in advance.

- Midsem - 25% - Online (Sep 27th, 2pm - 3:30pm).

- Endsem - 25% - Online (Nov 24th, 2pm - 3:30pm).

- Programming Assignments - 48% (6 Assignments, 8 Marks each) - to be submitted as jupyter notebooks through moodle.

- Among the remaining 2%, some part will be given for setting up the jupyter notebook. Some part will also be given for class participation.

6 Role of Teaching Assistants

The class is divided into five groups. Each group has a lead TA and a support TA. The TAs will be available for discussions on **Thursdays at 10am - 11am slot.**. The TAs may also contact their respective students in case of issues or corrective discussions are required regarding the assignment submissions. They may also suggest attending contact hours online on particular days to clear doubts.

7 Academic Honesty

Academic honesty is expected from each student participating in the course. NO sharing (willing, unwilling, knowing, unknowing) of assignment copy between students, submission of downloaded solution idea (from the Internet, Campus LAN, or anywhere else) is allowed. Discussions are allowed in solving the problem sets. Students are strongly encouraged to read this this description about what is plagiarism. Academic violations will be handled by IITM Senate Discipline and Welfare (DISCO) Committee.

Group	Students	Lead TA	Support TA
λ	Pre-2019 batch	Jayalal Sarma jayalal@smail.iitm.ac.in 9840920902	Nagashri K. CS21D004@smail.iitm.ac.in 8970839588
α	CS19B001 - CS19B020	Bhabya Deep Rai CS21S015@smail.iitm.ac.in 8436299969	Sutanay Bhattacharjee CS21D005@smail.iitm.ac.in 8812905399
β	CS19B021 - CS19B040	Michael Mervin Christy CS17B108@smail.iitm.ac.in 9789898014	Siddharth Dwivedi CS20M065@smail.iitm.ac.in 8057911389
γ	CS19B041 - CS19B060	Vimala S CS19D013@smail.iitm.ac.in 9894247216	Rituparna Adha CS20M054@smail.iitm.ac.in 8455992010
δ	CS19B061 - CS19B081	Deepali Ande CS20S052@smail.iitm.ac.in 8983389207	Pooja Kumari CS20D006@smail.iitm.ac.in 9455132445