

University of Utah
School of Medicine
Department of Biomedical Informatics



Semester: Fall 2020

Course Number: BMI 6018

Course Title: Introduction to programming

Instructor: Samir Abdelrahman – samir.abdelrahman@utah.edu

TA: RAMAMURTHY SIRIPURAM-- ram.siripuram@utah.edu

Credit Hours: 3

Office Hours: TBD with students

Zoom Invitation: Appendix A (Zoom Invitation) + the invitation was sent to each student.

Kick-off course page: To go through before your course start day on Aug 24th.

Prerequisite:

- **Go to:** <https://continue.utah.edu/coursera>. Register by your Uid(U0000000) for free. They will send you the log-in information.
- **Register, for free, the course: Programming for Everybody (Getting Started with Python):** <https://www.coursera.org/learn/python?specialization=python#syllabus>
- **Complete chapters one and two from the book:** http://do1.dr-chuck.com/pythonlearn/EN_us/pythonlearn.pdf. This should take approximately 4 weeks. The book could be downloaded from <https://www.py4e.com/book>. Please don't install the python as recommended by the course. You can skip these lessons or just read them for your own knowledge. It's best to use the course playground for now. Feel free to complete the entire course (this is optional).

University calendar dates: <https://registrar.utah.edu/academic-calendars/fall2020.php>

Description:

This course will equip students with programming skills to develop computer programs and to analyze publicly available datasets in the medical informatics domain. The students will learn python programming skills, their application using widely used platforms, and software version control using git. Mathematical and statistical principles relevant to biomedical data science will be reviewed through programming examples and problems.

Learning Objectives:

- 1- To understand programming procedural and object-oriented concepts.
- 2- To learn python commands and widely used platforms.
- 3- To practice algorithmic thinking for solving mathematical, statistical, and biomedical problems in python.
- 4- To apply python commands in real-life applications:

Materials and Media:

The course materials will be Jupyter notebooks. The instructor, TA, and students will use: (1) the School of Medicine (SOM) Jupyter GitHub server for their practice and (2) Slack, and Canvas to exchange any course information. The student has to understand every detail in Jupyter notebooks with all their embedding web links. The student has to read all materials in Canvas and Jupyter notebooks.

Textbooks:

- **There is no best book to learn a programming language".**
- **We will use the below books with bold titles [HTL-P3, P4E, MathP, and PDA]**

Book 1: "Think Python: How to Think Like a Computer Scientist 2nd Edition, Allen B. Downey, 2016".

- **Another version (HTL-P3): "How to Think Like a Computer Scientist: Learning with Python 3"**
 - o PDF version in the Canvas books folder
 - o Website: <https://www.ict.ru.ac.za/Resources/cspw/thinkcspy3/>
 - o Interactive version:
<https://runestone.academy/runestone/books/published/thinkcspy/index.html>

Book 2: "Python for Everybody (P4E): Exploring Data in Python 3, Charles Russell Severance, 2016"

- o PDF version in the course Canvas books folder
- o Website: <https://www.py4e.com/book>

Book 3: "Doing Math with Python (MathP): Use Programming to Explore Algebra, Statistics, Calculus, and More! 1st Edition, Amit Saha, 2015".

- o PDF version in the Canvas books folder
- o <http://index-of.es/Varios-2/Doing%20Math%20with%20Python.pdf>

Book 4: "Python for Data Analysis (PDA): Data Wrangling with Pandas, NumPy, and IPython, 2nd Edition, Wes McKinney, 2018".

- o Online old version: <https://bedford-computing.co.uk/learning/wp-content/uploads/2015/10/Python-for-Data-Analysis.pdf>
- o PDF version in the Canvas books folder

Others: Some many books and courses could be downloaded from websites. The instructor may use any content from any website when needed.

TA roles for the course students: The course TA roles are to help students only in debugging codes, installing python platform, and solving GitHub problems but not in understanding/coding any course question. Any other role will be determined by the instructor when needed. The TA office hours will be determined.

University Policies

The student should read the Canvas course university policies besides the below instructions.

The Americans with Disabilities Act. The University of Utah seeks to provide equal access to its programs, services, and activities for people with disabilities. If you will need accommodations in this class, reasonable prior notice needs to be given to the Center for Disability Services, 162 Olpin Union Building, (801) 581-5020. CDS will work with you and the instructor to make arrangements for accommodations. All written information in this course can be made available in an alternative format with prior notification to the Center for Disability Services.

Addressing Sexual Misconduct. Title IX makes it clear that violence and harassment based on sex and gender (which includes sexual orientation and gender identity/expression) is a civil rights offense subject to the same kinds of accountability and the same kinds of support applied to offenses against other protected categories such as race, national origin, color, religion, age, status as a person with a disability, veteran's status or genetic information. If you or someone you know has been harassed or assaulted, you are encouraged to report it to the Title IX Coordinator in the Office of Equal Opportunity and Affirmative Action, 135 Park Building, 801-581-8365, or the Office of the Dean of Students, 270 Union Building, 801-581-7066. For support and confidential consultation, contact the Center for Student Wellness, 426 SSB, 801-581-7776. To report to the police, contact the Department of Public Safety, 801-585-2677(COPS).

Code of Student Rights and Responsibilities: (Policy 6-400): The course students should understand and adhere to all rules (<https://regulations.utah.edu/academics/6-400.php>).

Cheating in the course will subject to the academic sanctions described in (<http://campusguides.lib.utah.edu/c.php?g=237735&p=1585547>). Cheating in the course includes all activities that the student copies or helps another student (s) to copy the materials, solution ideas, and codes of assignments, projects, and quizzes from any private or public resource. External libraries and source code can be used only for additional functionalities that are not core parts of the assignments/projects, and they should be clearly indicated and cited in comments within the source code and their reports. The instructor will have the right to apply any rule described in the mentioned two links for each cheating case, if any. Of note, the collaborations among the students from understanding the course materials and problems are encouraged, but **exchanging or sharing the course projects, and solutions of the course problems are prohibited and considered as cheating activities.**

If the student takes any idea or solution of any quiz/ assignment/project question from the TA/any other person/a web site, it will result in automatic zero for the whole related quiz/ assignment/project. If the student is stuck in understanding or solving any assignment question (s), he should immediately contact/email the instructor. The instructor will work to resolve the student' difficulties.

Tentative Course Schedule

This syllabus is not a contract. It is meant to serve as an outline and guide for your course. Please note that the instructor may modify it to accommodate the needs of your class. You will be notified of any changes to the syllabus.

Course Modules

1. Introduction to the course
 - a. Course overview and environment.
 - b. Python strength and weakness.

- c. Programming, algorithms, and flowcharts concepts.
- d. A crash course in python
 - i. Variables, expressions, and statements.
 - ii. A quick overview of python data structures and keywords.
 - iii. An introduction to functions and modules.
- 2. Learning python: Numeric and Boolean types
 - a. Integer numbers.
 - b. Float numbers.
 - c. Complex numbers.
 - d. Fractions and Boolean.
 - e. Digital number system representation: Hex, octal, and binary.
 - f. Order of operators and conversions among types.
- 3. Learning python: Conditional execution, iterations, and functions
 - a. If condition: Simple, compound, and nested.
 - b. Iterations: for and while loops.
 - c. Functions: built-in/def types, calls, return values, parameters, arguments, and flow of executions.
- 4. Learning python: Sequence and mapping datatypes
 - a. Strings, regular expressions, and text processing applications.
 - b. Lists.
 - c. Sets.
 - d. Tuples.
 - e. Dictionaries.
- 5. Learning Python: Advanced topics in functions, modules, and iterations
 - a. Scopes.
 - b. Recursion.
 - c. Anonymous functions.
 - d. Modules.
 - e. Packages.
 - f. Generators.
- 6. Learning python: Object-oriented programming
 - a. Class and object syntax.
 - b. Inheritance.
 - c. Operator Overloading.
- 7. Learning Python: Files and Exceptions
 - a. File commands, and CSV/JSON/free-text formats.
 - b. try/except/else/catch statements.
 - c. Designing exceptions.
- 8. Applications in python: Introduction to data analysis
 - a. NumPy, SciPy, Pandas, and Matplotlib.
 - b. Descriptive statistics.
 - c. Symbolic mathematics.
- 9. Applications in python: Introduction to database
 - a. SQL concepts.
 - b. SQLite interface overview.
- 10. Applications in Python: Introduction to the graph database
 - a. Concepts.
 - b. Neo4j interface overview.
- 11. Applications in python: Introduction to web application
 - a. Concepts.
 - b. Beautiful soup overview.

Date	Module	Chapters in Books ¹	Quiz	Assignment
Aug 24	1.(a-c)	Chapter 1 in HTL-P3 book and interactive version	Introductory Quiz	
Aug 26	1.d	Chapter 2 in P4E book		
Aug 31	1.d		Quiz 1	
Sept 2	2	Chapter 2 in HTL-P3 and interactive version		Ass 1
Sept 7	Labor Day Holiday – No class			
Sept 9	2, 3	Chapters 4,5,6,7 in HTL-P3 book		
Sept 14	3	Chapters 6,7,8 in HTL-P3 interactive edition	Quiz 2	
Sept 16	Appendix A in HTL-P3 book and other tutorials in Canvas Visual Studio Code crash course (TA) Project specification and discussion			Ass 1 Due Ass 2
Sept 21	4	Chapters 6, 8, 9, 10, 11 in P4E book	Quiz 3	
Sept 23	4			
Sept 28	4		Quiz 4	
Sept 30	5	Tutorials and web links in Canvas Chapters 12,18 in HTL-P3 book		Ass 2 Due Ass 3
Oct 5	5	Chapters 16 in HTL-P3 interactive version	Quiz 5	
Oct 7	5			
Oct 12	6	Chapters 15, 16, 18, 21, 22, 23 in HTL-P3 book	Quiz 6	
Oct 14	6			Ass 3 Due Ass 4
Oct 19	6		Quiz 7	
Oct 21	7	Chapters 13, 19 in HTL-P3 book		
Oct 26	7		Quiz 8	
Oct 28	8	Chapters 2,3,4,5 in the MathP book and some examples from PDA book		Ass 4 Due Ass 5
Nov 2	8		Quiz 9	
Nov 4	8			
Nov 9	9	Tutorials and web links in Canvas	Quiz 10	
Nov 11	9			Ass 5 Due
Nov 16	10	Tutorials and web links in Canvas		
Nov 18	10			
Nov 23	11	Tutorials and web links in Canvas		
Nov 25	No Class			
Nov 30	Uploading all projects before class time Project presentations with the instructor and TA			
Dec 2	Project presentations with the instructor and TA			

In the above table, the contents of the book' chapters of each module are the main directions and materials for the class lectures. These directions could be changed or adapted due to students' performance every week, which will be assessed by the instructor weekly. The specific materials for each lecture will be announced one week before the week of each lecture. The guidelines of

¹ Books' chapters are complementary materials to the instructor' lectures—The instructor will use them in the lectures and will provide them as reading materials to the students. Please go to the canvas to check the pre-class and in-class materials.

graphical user interface programming will be taught to the students during the course. Still, the students will implement some of these guidelines in their projects as self-learning tasks.

Course activities:

Quizzes (40% total—6 marks x 10 quizzes):

Students will have 10 quizzes with 6 marks or points per quiz. Each quiz will include, but not limited to, any question types of programming concepts and mathematical/statistical problem-solving in python. The students will do each quiz in the Canvas. Each quiz will be held in the last 10 minutes of the related quiz lecture. The syllabus and introductory quizzes will have no marks and it will be just to let the students know the course pre-knowledge and the structures of the other 10 quizzes; meanwhile, they let the instructor know the levels of programming knowledge and expertise of students before the course. Quizzes from 6 to 10 may follow a different format and different types of questions. At then, the instructor will provide essential background to students before these quizzes.

Assignments (40% total—12 marks x 5 Assignments):

Students will have 5 homework assignments with 12 marks per assignment. Each assignment will have only one student. The assignment will include, but not limited to, any question types of programming in python, mathematical problem solving, algorithm explanation, and the result analysis and interpretation. It will be designed to support the course learning objectives. The assignment due is shown in the course syllabus. The assignment grades will be announced after 1 week of its due in Canvas.

In each assignment, there will be a question that discusses new python concepts/commands. The students will interpret the related codes and apply them in a small use case. Also, debugging use cases will be other potential questions in all assignments.

Project (20%- 100 marks):

Each project will have at most two students. The students should use the instructions and the project ideas that will be announced in the September 16th lecture to do their projects. Each project students may email or Zoom the instructor during the semester to exchange thoughts and ideas with each other about the project. Also, the instructor will lead discussions through Canvas for each project. The final project submission materials should have extensions ".py" in a project folder that contains all modules and files, and that could be run in the Visual Studio Code. No Jupyter notebook will be accepted. The students could try their codes while they prepare their projects using Jupyter notebooks. All flowcharts and algorithms for the project modules should be submitted as well. According to the above table, each project should be submitted in Canvas before 5 pm on Nov 30. The project will be discussed with the instructor and the TA using Zoom in any time after 5 pm on Nov 30 or any time after 3 pm on Dec 2. Project discussion will take 20-minute. The discussion 20-minute time slot for each project will be determined and announced on Oct. 25th by the instructor. The project will have three parts:

- Part 1 (20 marks): the part is to implement a graphical user interface to the project application. This part is self-learning using the materials and the guidelines that the instructor will provide.
- Part 2 (30 marks): The flowcharts and algorithms for the project modules.
- Part 3 (50 marks): The codes of the project.

Policy rules in the course activities:

- There is no "drop the lowest score" policy for any course activity (i.e., quizzes, assignments, and projects)
- There will be no extra credit opportunities for any course activity.
- Grade queries must be submitted within one week of the homework solutions being posted.
- All student reports of each activity should be submitted through the Canvas system (no emails or any other medium).
- Late Policy: Each 1-day late will make the student loses 10% from the whole marks of the related activity. Consequently, 10-day late will result in automatic zero in the related activity. For example, if the student submits an assignment 1-day after the assignment due date, this will cause 1.2 marks to be deducted from the assignment 12 marks.

TA roles for the course students: The course TA roles are to help students only in debugging codes, installing python platform, and solving GitHub problems but not in understanding/coding any course question. Any other role will be determined by the instructor when needed. The TA office hours will be determined.

Appendix A (Zoom Invitation)

Samir Abdelrahman is inviting you to a scheduled Zoom meeting.

Topic: BMI6018-2020 (Introduction to programming)

Time: Aug 24, 2020 05:00 PM Mountain Time (US and Canada)

Every week on Mon, Wed, until Dec 2, 2020, 30 occurrence(s)

Aug 24, 2020 05:00 PM

Aug 26, 2020 05:00 PM

Aug 31, 2020 05:00 PM

Sep 2, 2020 05:00 PM

Sep 7, 2020 05:00 PM

Sep 9, 2020 05:00 PM

Sep 14, 2020 05:00 PM

Sep 16, 2020 05:00 PM

Sep 21, 2020 05:00 PM

Sep 23, 2020 05:00 PM

Sep 28, 2020 05:00 PM

Sep 30, 2020 05:00 PM

Oct 5, 2020 05:00 PM

Oct 7, 2020 05:00 PM

Oct 12, 2020 05:00 PM

Oct 14, 2020 05:00 PM

Oct 19, 2020 05:00 PM

Oct 21, 2020 05:00 PM

Oct 26, 2020 05:00 PM

Oct 28, 2020 05:00 PM

Nov 2, 2020 05:00 PM

Nov 4, 2020 05:00 PM

Nov 9, 2020 05:00 PM

Nov 11, 2020 05:00 PM

Nov 16, 2020 05:00 PM

Nov 18, 2020 05:00 PM

Nov 23, 2020 05:00 PM

Nov 25, 2020 05:00 PM

Nov 30, 2020 05:00 PM

Dec 2, 2020 05:00 PM

Please download and import the following iCalendar (.ics) files to your calendar system.

Weekly: <https://lms-utah.zoom.us/meeting/tJEpd-uvqD8pH93LhjrEuhF3y0sJJ6W4TQqc/ics?icsToken=98tyKuCtqjsoHdKRszyORowMBoiga-3wpilegvoOsBfvEQVGaiWIL7JRGJgpJ4zk>

Join Zoom Meeting

<https://lms-utah.zoom.us/j/95402764359>

Meeting ID: 954 0276 4359

Passcode: 813908

One tap mobile

+12532158782,,95402764359# US (Tacoma)

+13462487799,,95402764359# US (Houston)

Dial by your location

+1 253 215 8782 US (Tacoma)

+1 346 248 7799 US (Houston)

+1 669 900 6833 US (San Jose)

+1 301 715 8592 US (Germantown)
+1 312 626 6799 US (Chicago)
+1 929 205 6099 US (New York)
Meeting ID: 954 0276 4359
Find your local number: <https://lms-utah.zoom.us/j/akQhjEZCo>

Join by SIP
95402764359@zoomcrc.com

Join by H.323
162.255.37.11 (US West)
162.255.36.11 (US East)
115.114.131.7 (India Mumbai)
115.114.115.7 (India Hyderabad)
213.19.144.110 (EMEA)
103.122.166.55 (Australia)
209.9.211.110 (Hong Kong SAR)
64.211.144.160 (Brazil)
69.174.57.160 (Canada)
207.226.132.110 (Japan)
Meeting ID: 954 0276 4359
Passcode: 813908