
Learning Outcomes

In this course, you will be invited to develop your ability to think clearly and carefully about data structures and the algorithms that operate on them, and to improve your skills in expressing those thoughts about data structures in a precise way. Data structures are a crucial component of most computer applications. Understanding them is a key ingredient for writing correct and efficient computer programmes. By the end of this course, you will be able to do the following things.

- **Describe important classes of data structures (dictionaries, priority queues, disjoint set union) and explain how they work.**
  Before trying to design your own data structures, it is important to be aware of a good collection of classical ones. They are useful in their own right, but the ideas that underlie them are also useful for new data structures.

- **Augment data structures to expand their functionality.**
  If one of the standard data structures doesn't work for the task at hand, it is useful to be able to tweak an existing one to fit your purpose.

- **Analyze the efficiency of data structures, including the use of amortized analysis and online competitive analysis.**
  In many cases, there are multiple data structures that could be used to solve a problem. In some cases, it is important to be able to analyze the different options and choose the one that will solve the problem most efficiently. Amortized and competitive analysis are two special types of analysis that are useful in some settings.

- **Demonstrate knowledge of important themes in data structure design such as persistence, self-adjustment, concurrency.**
  This course is not just about going through a catalogue of data structures. Certain ideas can be applied in the construction of many different data structures, and we want to understand those concepts too. Persistence
deals with maintaining multiple versions of stored data so that old versions can be accessed when needed. Self-adjustment is used instead of carefully rebalancing data structures to improve the shape of the data structure in response to the sequence of operations performed on it. Concurrent data structures allow multiple threads to access the data simultaneously.

- Select or design appropriate data structures to help solve problems for algorithmic applications. The ultimate goal is to ensure that you can find (or build) the right data structure for whatever task you want to solve. You should also be able to justify your design choices and explain why the algorithms that operate on the data structures are correct.

How to Learn This Material

Much of the material in this course is in the same vein as the material of EECS3101. To develop your understanding of the material, it is important to do more than just read the textbook and attend lectures: you should work through exercises.

You can often learn by struggling with problems. However, if you get too stuck or don't know how to begin, help is available. Talk to your classmates (however; see the notes below about academic honesty regarding discussing assignment problems with others). Go to office hours; the instructor is there to help you! You also learn by making mistakes and getting feedback about those mistakes. Just make sure that you use the feedback to improve your understanding.

Groups of students can learn a lot by explaining their solutions to the suggested exercises from the textbook to one another and critiquing the solutions of others. After all, learning how to explain solutions clearly is one of the goals of this course. Seeing where other students' solutions are unclear to you helps you make your own explanations clearer. Be aware that a problem may have many different correct solutions; just because someone's solution is different from yours doesn't necessarily mean that one of them is wrong.

It takes time to build new skills, so it helps if you work on exercises regularly: don't leave all the work to the days right before a test. Similarly, some of the homework assignments will be difficult to finish if you leave them to the last minute.

EECS 4101 Marking Scheme

- Homework exercises - 15%
- Test #1 - 22.5%
- Test #2 - 22.5%
- Exam - 40%

It's a very good idea to type your solutions to homework assignments, since it allows you to edit and polish the answers, but handwritten solutions are also acceptable, as long as they are legible. If you want to type your solutions, LaTeX produces elegantly typeset documents, is available for free, and was built to handle even the most complicated mathematical notation. It can take a while to learn how to use it, but once you do, you will probably not want to type documents any other way.

You should make every effort to make your answers as brief as possible, while still being thorough. Brevity requires careful thought and editing. (Pascal once excused himself for writing a long letter, saying that he did not have enough time to write a shorter one.) Students who write copious amounts usually do not know what they want to say, or are saying it in a very disorganized way. Usually, an answer to a homework question should fit on one sheet of paper. If you are writing much more than that, you probably have not found the best way to solve it. On tests, your answer should usually fit into the space provided for it.

Textbook


Other References

• Pat Morin, *Open data structures*.
• Clifford A. Shaffer, *Data Structures and Algorithm Analysis*, edition 3.2.0.10, 2013.

**ACADEMIC INTEGRITY LINKS**

- Senate Policy on Academic Honesty - http://secretariat-policies.info.yorku.ca/policies/academic-honesty-senate-policy-on/
- Academic Integrity - http://lassonde.yorku.ca/academic-integrity

**STUDENT LINKS**

- Student Rights and Responsibilities - http://oscr.students.uit.yorku.ca/student-conduct
- Religious Observance - https://w2prod.sis.yorku.ca/Apps/WebObjects/cdm.woa/wa/regobs
- Counselling and Disability Services - http://cds.info.yorku.ca/
- York University’s Policies on Gender/LGBTQ*/Positive Space - http://rights.info.yorku.ca/lgbtq/

**LAND ACKNOWLEDGEMENT**

- We acknowledge our presence on the traditional territory of many Indigenous Nations. The area known as Tkaronto has been care taken by the Anishinabek Nation, the Haudenosaunee Confederacy, the Huron-Wendat, and the Métis. It is now home to many Indigenous Peoples. We acknowledge the current treaty holders, the Mississaugas of the New Credit First Nation. This territory is subject of the Dish With One Spoon Wampum Belt Covenant, an agreement to peaceably share and care for the Great Lakes region.
- The Indigenous Framework for York University: A Guide to Action can be found here: http://indigenous.info.yorku.ca/
- Meaning of a land acknowledgement: http://healthydebate.ca/opinions/indigenous-land-acknowledgements

---

Many courses utilize Moodle, York University’s course website system. If your course is using Moodle, click here to access it.

*Moodle @ York University*