Syllabus for Algorithms Fall 2019

People

- Instructor (**until the Fall break**): Prof. José Bento. **Tip:** To maximize your chances of reaching me use class time, CANVAS messaging, and office hours, where I can meet you in person, individually or in groups. I receive many emails, so using email is a bad method to grab my attention. **After the Fall break** Prof. Lewis Tseng (lewis.tseng@bc.edu) will take over the class. Prof. Bento will be on leave.
- TAs: Michelle Youn (younmi@bc.edu) and Helen Nazarenko (nazarenh@bc.edu).
- Extra support: **Tip**: The Connors Family Learning Center offers great tutoring services on a variety of CS related topics. I encourage you to use them if necessary. Link

Office hours

- Prof. Bento (**until Fall break**): After class, 6:15pm-7:30pm, MW, Saint Mary's Hall South, S250. I will try my best to meet with you outside this schedule if you reach out to me via CANVAS, but I cannot guarantee that I will be available.
- Prof. Tseng (**after Fall break**): TBD and by appointment, Saint Mary's Hall South, S273.
- TAs: Helen Nazarenko, Mon 6pm-8pm; Michelle Youn, Wed 10pm-12:00am; Fulton 160

Class schedule

Wednesday and Friday: 4:30pm to 5:45pm, 310 Gasson Hall. During lecture time **no cellphones/computers/tablets/players/etc, and no food, are allowed**. You are, of course, very welcome to step outside for a snack, phone call, or a break, and them come back. Remember that you do not have to be in the lecture room, but if you willingly want to be there, you need to respect these minimal set of rules.

Prerequisites

Being fluid in

- Calculus (to the level of Calculus I)
- Probability (to the level of Randomness and Computation)
- Automata (to the level of Logic and Computation)

Unfortunately, having taken and forgotten the content of these courses does not fulfill the prerequisites. I strongly recommend you to refresh your memory. I try to make all of my lectures as self-contained as possible. **Tip**: You are encouraged to ask questions about prerequisites during lectures, or in office hours. You can also use the TAs, or the tutoring services aforementioned, to refresh those prerequisites. However, this class is not a class on calculus, probability, or automata theory, which I assume you have already mastered. Unfortunately, I cannot devote most of my time to teach you these topics from scratch, at the expense of not teaching algorithms to the students that do have the required background, and have the right to see the class topics covered in class. To help each student self-assess if he is ready to take this class, I will give a prerequisite quiz in the first lecture. Your score in this quiz **will not** count towards your class grade. Your score will help you self-assess if you are ready to take this class this year. There is nothing wrong in delaying one year to take this class.

If you understand everything that is reviewed in class sessions and you can apply all of those skills on the homework problems, you will probably do well in the class. However, you might be asked to extrapolate the main techniques and ideas learnt in class to examples or problems that you have not seen before, just like when learning e.g., English, one learns some specific words and reads some specific texts in class, but then combine these "atoms" to produce new texts.

Reading

Books are expensive, and not everyone can afford them. There is no mandatory book for the first half of the class. I try to make all of my lectures as self-contained as possible (prerequisites assumed). However, for those that do want to read a book, there are many good books on algorithms. A good and comprehensive one is *Introduction to Algorithms* (IAC) by Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein, SBN 978-0-262-03384-8. Link

After the Fall break, Prof. Lewis Tseng will follow *Algorithms* by S. Dasgupta, C. Papadimitriou, and U. Vazirani. Algorithms, McGraw-Hill, SBN 978-0-073-52340-8. Link A free draft version can be found here (**Tip:** Replace "chap0.pdf" in the URL by "chap1.pdf", "chap2.pdf", etc. to access other chapters).

Course topics

In the first part of the semester, the level of detail with which I will cover the different class topics will vary with students' interests and time available. For each topic listed below, I include a reference to the relevant pages in the book *Introduction to Algorithms* (IAC), mentioned above for those that want to do extra reading. Note that the book might not cover everything that I teach in class, or the same way that I cover it. It often goes beyond what is covered in class. To guarantee that I cover everything in the syllabus, sometimes I might record a lecture and put it online for student viewing. **Tip**: Use the list of objectives in the table below self-assess your learning. Note: these objectives are guidelines, and are the minimum that each student should grasp for each topic. There is more to each topic than what can be covered in a syllabus table. A high-achieving student should feel comfortable in (i) establishing connections between the different topics, not see them in isolation, (ii) applying the main ideas in each topic beyond the examples covered in class, (iii) solving the same problem in several different ways (when possible).

After the Fall break, i.e. starting at Week 9, the flow of the class will follow Prof. Lewis Tseng's syllabus, which you can find here. Prof. Bento will be on leave.

Week	Topics	Pages in IAC	Due dates	Objectives
1	• Course overview and syllabus; • Self-assessment quiz; • Something that algorithms cannot do	181-188	HW0 26th Aug - -2nd Sept	Give examples of tasks that a Turing Machine cannot perform.

2	• Sorting numbers
	  ○ Merge sort
	  ○ Sorting networks
	  ○ Quick sort
		29-37
		719-724
		170-184
			HW1
			2nd Sept—9th Sept
				Calculate lower bounds on performance. Implement sorting algorithms. Explain their merits and

				limitations. Explain and use recursion.
3	• Sorting numbers ○ Quick sort (finish) • Hashing	170-184 253-269	HW1 9nd Sept— 16th Sept	See above. Explain why hashing is useful.
4	• Binary heaps	151-164	HW2 16th Sept— 23rd Sept	Implement binary heaps. Analyze their performance.

5	• Complexity analysis
	  ○ Order notation
	  ○ Solving recurrences
		43-60
		83-106
			23rd Sept -
			30th Sept
				Formally define the different types of order notation. Express time and memory complexity in order notation. Derive recurrences for the performance of different algorithms. Solve recurrences to derive upper and lower performance bounds.

6	• Complexity analysis
	  ○ Solving recurrences (finish)
	• Dynamic Programming
	  ○ Paragraph breaking
	  ○ Shortest path
		83-106
		359-370
			HW3
			30th Sept—7th Oct
				See above. Explain how dynamic programing works. Use it to solve simple problem. Implement solutions to the three problems discussed in class.

7	• Dynamic Programming
	  ○ Shortest path (finish)
	  ○ Towers of Hanoi
		359-370	HW3
			7th Oct—
			14th Oct
				See above.

Fall break, 14th and 15th of October. Prof. Lewis Tseng takes over after the midterm. See his syllabus here. Below is what Prof. Bento would normally teach, just for reference, and complementarity.				
8	• **Midterm I**		16th Oct	

9	• Graph algorithms
	  ○ Graph representation
	  ○ Breadth first search and depth first search
		589-592
		594-620
			HW4
			21st Oct—28th Oct
				Explain the merits and limitations of expressing graphs using different data structures, including, adjacency matrices and lists.
				Give examples of applications of why BFS and DFS are important.
				Implement and analyze BFS and DFS.

10	• Graph algorithms ○ Minimum spanning trees	624-636	28th Oct—4th Nov	Give examples of why the MST problem is important. Implement and analyze Prim's and Kruskal's algorithms. Explain the main idea behind greedy algorithms.

11	• Graph algorithms
	  ○ Minimum spanning trees (finish)
	  ○ Maximum flows
		624-636
		708 - 730
			HW5
			4th Nov—11th Nov
				See above. Give examples of why the MF problem is important. Implement and analyze Ford-Fulkerson's and Edmonds-Karp's algorithms.

12	• Graph algorithms ○ Minimum cut/ maximum flow theorem	723-724 732-735	HW5 11th Nov—18th Nov	Use the min. Cut/max. flow theorem to solve problems. Reduce the

	○ Assignment problem			assignment problem to a max. flow problem and solve it.
13	• Discrete Fourier Transform ○ Complex numbers ○ Fast Fourier Transform	1-19 906-912	HW6 18th Nov— 25th Nov	Perform basic algebraic operations with complex numbers. Define and use the discrete Fourier Transform and its inverse. Implement the FFT algorithm. Analyze its complexity.
14	• Discrete Fourier Transform ○ Fast Fourier Transform (finish)	906-912	HW6 25th Nov—2nd Dec	See above.

Grading

- 36% homework
  - Homework consist of written and programing assignments, including implementing algorithms discussed in class. These assignments will be posted on CANVAS, where you can see their deadline.
  - No extensions or late homework is accepted. To compensate this tough policy, the homework with the least score will be dropped in calculating the homework average.
  - In extreme circumstances, we may forgive homework. Extreme circumstances include serious illness, or injury. Please contact the instructor if these happen.

    In the first half of the semester…
  - Assignments will link you to Hacker Rank. Hacker Rank is a website where you will receive a list of programing tasks, upload your code, and will have your code's behavior tested on a set of pairs of input-output files. Your grade on each question of the assignment will be proportional to the number of input-output tests that your code can successfully pass. The more tests your code passes, the higher your grade. For each homework, your grade will be a value from 0% to 100% obtained from the fraction of the number input-output tests that your code passed over the total number of input-output tests that there are.

The deadline for each assignment will be posted on CANVAS. Hacker Rank will also only be available for the duration of each assignment. In addition to submitting your code to Hacker Rank, you are also to submit your code in a zip file in CANVAS, as explained in each CANVAS assignment. The grading is all done automatically by Hacker Rank.

- If you have questions about using Hacker Rank, problems uploading files, or other administrative issues about homework, please contact the TA's about this, not me. On my hand, I am more than happy to help with the homework, conceptually. However, I am not going to solve it or code it for you, or debug your code, or discuss homework administrative issues with the students. The reason for this is that the TA will be the ones dealing with Hacker Rank, bookkeeping all grades, and making sure homework goes online, and offline, at the correct deadlines.

After the Fall break…

- There will be a couple of hand written homework (roughly one per two week).
- This Homework can be worked on in groups of up to 3, and each group submits one written solution.
- Your assignment must be typed and submitted electronically. Typing an assignment with a lot of math in it can be a chore, but electronic submissions of photographs of handwritten have usually been difficult to read. There are two methods that you can use to prepare the assignment:
  - Use the equation editor in Microsoft Word.
  - Use LaTeX, like you have done so for the scribed lectures.
- Please see more instructions about HWs after the Fall break here, or contact Prof. Tseng if you have questions.

- 30% midterm
  - There will be two midterm exams in this class. One on the 16th of October (prepared and graded by Prof. Bento), and one on the 6th of November (prepared and graded by Prof. Tseng). These dates are subject to changes depending on the speed of the class. Each midterm exam will cover all of the topics discussed in class, and covered in the homework, right up to the exam's date. Prof. Bento's exam is to be hand solved/written, with no access to books, notes, computers, cellphones, or any other kind of auxiliary source of information. For Prof. Tseng's exam, you are typically allowed **one page of notes**. Please confirm with him before the second midterm.

- 14% scribing
    - Each two students will be required to scribe a lecture on Overleaf using Latex. The link to the Overleaf document is this Link. Each pair of students has a folder and latex file that only them should edit. There is version control, so I know what each person is doing, and mistakes can be reversed. But please do not break the whole document, or others' work. There will be a CANVAS assignment for each scribing assignment that only the assigned students should complete. Each pair of students should scribe their lecture within 1 week of the lecture taking place. Once the lecture is scribed in Overleaf, the students should also submit a Pdf version of the latest version of the document to CANVAS through the assignment created for him.
    - **Tip:** Look at last year's scribing (Link) to understand how to use Latex. Do not use it for any other purpose than to see how certain Latex features are used. Copying material from last year's scribing is a violation of integrity, and will be severely punished. To get a good score in this grade component, your scribed lecture should be substantially better than the equivalent lecture from last year, specially taking into account that you have last year's Latex code as a starting point.
    - The main purpose of this exercise is to teach you how to write in a rigorous and clear way, to learn Latex, and to learn how to use collaborate editing (which is the future). The notes can also be a good way to study, but remember that you should take your own notes in class, and not trust that what other students wrote is correct. Do not trust that last year's scribing is correct either.

- 20% project
    - There will be no final exam (:D) but a final project, which Prof. Tseng will direct. Although he only takes over the class after the Fall break, **choose your topic early and start working on it early.** Discuss your topic with Prof. Lewis as soon as possible. **Tip: You are welcome to go to his office hours and discuss project topics with him.** Prof. Tseng will be the one grading the final project. You can see a list of potential topics here, but you are free to suggest other topics. You are welcome to discuss your project with Prof. Bento as well, even though he will not be grading it and will be on leave after the Fall break.

Integrity

Do not copy your solutions for the homework from online sources or your colleagues. If you get caught, you will get into serious trouble. It is OK if you cannot solve a problem. Because of this, be careful with your collaborations. It is Ok to ask for help from a conceptual point of view. However, if you ask for very detailed help, there will be almost-equal homework solutions, and you will hurt yourself and your helping-friend. For exams and midterm, absolutely no information exchange is allowed.

Please read the following:
https://www.bc.edu/offices/stserv/academic/integrity.html