CSCI 4104: Data Structures & Algorithms II
Fall 2020 Syllabus

Instructor: Dr. Vincent A. Cicirello   **Office:** G116
E-mail (course related): Use course Blackboard Course Messages tool.   **Phone (office):** x3526
E-mail (other than this course): See campus directory for address.

Course Time and Location: The course is **Online**. The course is listed in the schedule as "Online Blended," which technically means partially synchronous with specific meeting times, and partially asynchronous. **However, the specific meeting time of Tuesdays, 10:30am-12:20pm is optional**. I will post a link to Zoom for this in Blackboard. You are not required to attend the Tuesday synchronous meeting. If you do attend, you should be prepared to ask questions. The purpose of this optional synchronous meeting is to answer questions that you have about homework problems, programming assignments, or just the course content more generally. I'm not going to take attendance during these Zoom meetings. And if nobody has any questions for me, then they will be over rather quickly. Also, if none of you are connected to the Zoom meeting by 10:45am, I will simply end the meeting.

Office Hours: Zoom by appointment (contact me via Blackboard course messages tool).
- The reduced room capacity to account for social distancing during the pandemic for my office is 1 (i.e., if I'm in there, nobody else is allowed to be). So I'm not allowed to meet with you there.
- If the reason for meeting with me is something general enough to be relevant to the class as a whole (e.g., you are stuck on how to do a homework problem, or would like me to do another example of how an algorithm works, etc), then I'd prefer that you start with the **Optional Tuesday Synchronous Class Meeting** (see "Course Time and Location" section above). Your question may benefit others if asked at that time rather than during virtual office hours.
- **Scheduling Virtual Office Hours**: Since office hours will be virtual via Zoom, I'm not setting specific days/times. Just send me a message via the Course Messages tool in Blackboard. For simple issues, we might be able to simply interact that way. But if you want to "meet" via Zoom include some suggested days/times (multiple options preferably) in your message, and I will either pick one or I'll suggest some alternatives if your requested days/times conflict with my schedule. Note: I won't meet on weekends or holidays, or after 4:30pm (I often have regular meetings for committees, etc at 4:30, and I don't work in the evening). My preference for meetings with students is between 10am and 2pm, but can be available outside that time window if necessary.
- You can call my office phone. I will be there some days/times, and even if I'm not, I will get any voicemail you might leave. Although if I don't answer when you call, you will likely get a quicker response if you send me a message in Blackboard.

Electronic Communications:
- **Discussion Forum:** I've set up a threaded discussion forum within Blackboard. If you have any general questions about course content, homework, exams, etc, feel free to post in that forum. You can also feel free to discuss course related topics with your classmates there as well. There are no requirements to post in the discussion. It is there as an additional way to communicate with your classmates, and me. If someone posts a question about something, and you happen to see it before I do and think you know how to answer, you can even answer each other's questions there. **Rule 1:** You are not allowed to post your solutions (or parts of solutions) to homework or exams. **Rule 2:** You are not allowed to post your own personal grade details (this should be obvious since the entire class can read the contents of the discussion forum). **Rule 3:** You must be respectful with each other (this should also be obvious).
- **Course Messages:** If you do have something specific to you (e.g., your grades, or if you want to set up an office hours appointment), then use the "Course Messages" link in Blackboard.

- **My Response Time:** If you post in the Discussion Forum and your post is something that I must respond to, or if you send me a direct message in Course Messages, I will respond no later than 11:59pm the next weekday. E.g., if you post/message anytime on Monday, I will reply by 11:59pm on Tuesday night, or if you post/message anytime on Friday, Saturday, or Sunday, I will respond no later than 11:59pm on Monday night. Thanksgiving and the Friday following Thanksgiving are treated like weekend days. I'm also highly unlikely to respond on the two precepting days.

Course Description: In this course, students deepen their knowledge of the design and analysis of computer algorithms. Advanced topics in algorithms and algorithm analysis covered in the course include graphs and graph algorithms, string matching, multi-threaded algorithms, and NP-completeness.

List of Course Topics:
- Introduction to algorithms
- Python programming
- Algorithm analysis (both asymptotic analysis and amortized analysis)
- Graphs (data structures and basic algorithms)
- Minimum spanning tree algorithms
- Single source shortest paths algorithms
- All pairs shortest paths algorithms
- Multithreaded (or Parallel) Algorithms
- Multithreaded and multiprocess programming in Python
- String matching algorithms
- NP Completeness

Prerequisites:
- CSCI 3103: Data Structures & Algorithms I (C or better),
- CSCI 2226: Foundations of Computer Science (C or better), and
- MATH 2215: Calculus I (C or better).

This course is a Q2 (Quantitative Reasoning Across the Disciplines): Among the math concepts from prior courses that you will use in this course are the following: (a) discrete math topics including set theory, logic, graphs, trees, and several other topics covered in MATH 2225/CSCI 2226; and (b) calculus topics including limits and derivatives (you might want to dust off your calculus textbook if you still have it).

Required Textbooks/Readings:
- Introduction to Algorithms, 3rd Edition, 2009, by T.H. Cormen, C.E. Leiserson, R.L. Rivest, and C. Stein. ISBN: 9780262033848.

Useful/Suggested Resources:
- Your calculus textbook (if you still have it).
- Your discrete math textbook (if you still have it).

Other Required Resources:
- Python 3 (current version is 3.8.5, but any version that starts with 3 is sufficient): https://www.python.org/
- **Python 3 will be used for all programming in this course.** [Note: Python 2 is still actively used but it is no longer maintained, and more importantly it has differences in syntax and different libraries than Python 3. Make sure you use Python 3. If you submit programming assignments that require Python 2, you will get a 0 for those assignments. Python 2 is now permanently frozen as of April 2020.]

- Useful python related links:
  - Tutorials: https://docs.python.org/3/tutorial/
  - Language and API documentation: https://docs.python.org/3/
- **The Respondus Lockdown Browser and a WebCam**: The exams for this course will be taken within Blackboard. However, you are required to take them using the Respondus Lockdown Browser on a computer with a WebCam (either one built into the computer such as a laptop WebCam, or an external usb WebCam).
  - Respondus uses your WebCam to automatically proctor the exam.
  - The Lockdown Browser gives you access only to the exam in Blackboard and nothing else (e.g., no other web browsing, no printing, no screen captures, no other applications).
  - Follow the instructions in Blackboard to install the Lockdown Browser. I also have a fake test set up in Blackboard that you can access an unlimited number of times to verify that Lockdown and Respondus is operating correctly.

Stockton Computer Science Student Learning Outcomes: This course supports the following CSCI Student Learning Outcomes:

- 1. An ability to analyze a complex computing problem and to apply principles of computing and other relevant disciplines to identify solutions.
  - a. Students will analyze a complex computing problem.
  - b. Students will apply principles of computing and other relevant disciplines to identify solutions to a complex computing problem.
- 2. An ability to design, implement, and evaluate a computing-based solution to meet a given set of computing requirements in the context of the program's discipline.
  - a. Students will design a computing-based solution to meet a given set of computing requirements.
  - b. Students will implement a computing-based solution to meet a given set of computing requirements.
  - c. Students will evaluate a computing-based solution to meet a given set of computing requirements.
- 3. An ability to communicate effectively in a variety of professional contexts.
  - a. Students will write technical documentation of a computer-based system, process, component, or program.
- 6. An ability to apply computer science theory and software development fundamentals to produce computing-based solutions.
  - a. Students will apply computer science theory to produce computing-based solutions.
  - c. Students will evaluate the effects of alternative data representations and algorithms on the performance of computing-based solutions.

Course IDEA Objectives: The IDEA objectives of this course (paraphrased within course context) are:
- IDEA Objective 1: Gaining knowledge of the terminology, methods, trends of data structures and algorithms, including the fundamental principles and theories of algorithm analysis, such as asymptotic analysis, and complexity classes.
- IDEA Objective 3: Learning to apply data structures and algorithms to solving real-world problems.
- IDEA Objective 4: Developing professionally relevant skills, such as programming in the Python language, parallel programming concepts, among others.
- IDEA Objective 11: Learning to analyze and critically evaluate alternative data structures and algorithms.

Course Content: All course content is within Blackboard. The **"Topics (Videos and Notes)"** page is where you will find all of the course topics. For each topic, I have recorded several videos, as well as have provided in note form any PowerPoint slides I used during the videos. You can also find Python source code used in the videos there as well. If you don't watch the videos, then you won't do well in the course.

Grading: Written Problem Sets (6)   20%
                Programming Assignments (5)   35%
                Exams (2)   45%

Grading Scale:

A: at least 90.00	A-: at least 89.00	B+: at least 88.00
B: at least 80.00	B-: at least 79.00	C+: at least 78.00
C: at least 70.00	C-: at least 69.00	D+: at least 68.00
D: at least 60.00	D-: at least 59.00	F: less than 59.00

I reserve the right to adjust the scale at the end of the semester. Such adjustments are rare, but will only be in your favor; and are highly unlikely to occur at the D-/F boundary. Note the 2 decimal places in the chart above (i.e., I do not round to the nearest whole number): e.g., unless I adjust the grade scale, an 89.99 is an A-, etc. If I adjust the scale, it is done using a semi-automated approach involving clustering (i.e., "automated" == a program I wrote suggests a new scale based on all of the grades of the class; "semi-" == if that program's output is crazy, I ignore it and leave the scale alone; and "clustering" == a statistical technique). I never simply add a constant number of points to everyone's overall course score.

Other Grade Related Info: If you are in the B.S. in Computer Science, then this course is a core required course and must be passed with at least a C to count toward your degree requirements. If you are in the old B.S. in CSIS (CS concentration), then this course is just a CS elective and must simply be passed (any passing grade).

Exams: The exams are not explicitly cumulative, although many concepts covered in the course build on earlier topics. You are allowed to use paper notes and the textbook during the exams. You are not allowed to use any other resources during exams (no electronic devices other than the computer on which you are taking the exam, which includes no cell phones, no tablets, no second computer---see academic honesty policy later in syllabus for penalty details). Respondus is used to ensure this rule is followed. There will be no Python syntax or programming on the exams. However, there may be problems on the exams involving reading and/or writing algorithms in pseudocode.

Exam Timing: The dates of the exams can be found in Blackboard. For each exam, there is a 48 hour time window. During that time window, you can choose any 3 hour block to take the exam. Once you start the exam, a timer starts enforcing the 3 hour time limit. The exams are not designed to require 3 hours. They are actually designed assuming a 2 hour block. E.g., they are approximately what I'd have you do on paper in a classroom if we had a two hour class module. I consider 1 of those 3 hours to be a buffer to deal with any technical issues that may come up during the exam. If you lose your Internet connection during the exam (either on purpose or due to issues beyond your control), then the timer continues to count down (it doesn't stop). In this case, you will need to reconnect as soon as you can resolve your issue. Blackboard will allow you to continue from where you left off assuming you haven't run out of time. In nearly all cases of technical issues, the problem is usually resolve within a few minutes, so a 1 hour buffer is ample time to deal with this. Unless you are able to document extensive technical issues lasting over 1 hour, you won't be given any extra exam time for such circumstances.

Make-Up Exams: Make-up exams will not be given (i.e., missed exam = 0), with the following exceptions:
1. Medical excuse: Provide documentation, preferably to the Wellness Center who will then contact all of the instructors of your courses.
2. Based on University policy (https://stockton.edu/policy-procedure/documents/procedures/2030.pdf), if you are to be absent for a religious holiday on the date of an exam, you must notify me of that planned absence during the first 10 business days of the semester. Keep in mind though that the religious holiday would need to span both days of the 48 hour time window of the exam to be relevant (if the holiday is only during one of the two days, then you can take the exam the other day).

Programming Assignments: There will be 5 programming assignments. The time allotted to each will generally be 2 weeks, but may vary depending upon the amount of work required. You will be required to work independently. Programming assignments will involve implementing data structures and algorithms covered within the course, and either applying them to a given problem, or in some cases performing an experimental comparison of alternative algorithms for a problem. Programming assignments will be implemented in the Python 3 language. Any assignments submitted that require Python 2.7 will receive a grade of 0. Details of all programming assignments, including deadline are available in Blackboard from the start of the semester. You are free to work ahead if you wish.

Written Problem Sets: Written problem sets will include problems, exercises, and review questions related to the course topics. The written homework exercises must be done individually. You must show your work for full credit. A simple answer without showing the work that lead to it will receive no credit in most cases.
- **Paper than Scan**: I strongly suggest that problems involving math are done on paper and then scanned to submit in Blackboard. It will probably take you less time to do it that way, since you need to show all of your work, then to try to format math in Word or some other word processing software.
- **Phone PDF Scanner Apps**: If you don't have a scanner, there are some good (and free) apps available for phones that use your phone camera to scan a document as a pdf (including multipage documents). One such app that I have personally used on Android is "Office Lens" from Microsoft. It is free and even has no ads (some of the other free aps have ads).
- Yes, I am aware that there are websites containing the textbook solutions. A few graduate students at a school I won't name worked through all of the problems and posted their solutions online. I have a copy of this, so I will know if you use them. Please see the academic honesty policy in the syllabus. Also, approx. 33% of their solutions are either totally or partially incorrect; and 33% are incomplete (e.g., an answer without work, etc). So even if I don't notice, you won't likely get a good grade using those solutions.

Due Dates: Assignments are due in Blackboard by 11:59pm on the dates due (dates found in Blackboard on the assignment pages). Late assignments are penalized as follows: (a) 25% off if late by no more than 24 hours, (b) 50% off if late by no more than 48 hours, (c) 75% off if late by no more than 72 hours, and (d) a grade of 0 if late by more than 72 hours. The first time an assignment of each type is late (within 72 hours), the late penalty is waived—i.e., you can be late with one programming assignment (within 72 hours of deadline) AND one written problem set (within 72 hours of deadline).

Academic Honesty: Please familiarize yourself with Stockton's policy on academic honesty. Each violation is penalized by a 0 on the relevant assignment/exam/etc, plus a 10 point penalty on your overall course grade. For example, if you have one violation, you'll have a 0 on that assignment or exam plus 10 points off your overall average, but if you have two violations, you'll have grades of 0 on the two assignments/exams/etc and 20 points off your overall average. Example violations include, but are not limited to: (a) any form of cheating on an exam or assignment, (b) passing off the work of another as your own (including other students, former students, code or problem solutions found on the Internet written by someone else, etc), (c) assisting someone in violating the academic honesty policy, (d) asking someone to

assist you in cheating or other academic honesty violations (even if they refuse to help you cheat), etc. [Yes, I encountered that last one once in a General Studies course.]

Incomplete Policy: In general, no grades of incomplete will be given. The only exception to this rule is an institutionally documented medical emergency that necessitates your complete absence from Stockton for at least two continuous semester weeks. Additionally, you must be caught up on all work up to the point where your medical emergency began and currently in the "C" range or better overall at the point where the emergency began.

LockDown Browser + Webcam Requirement: This course requires the use of LockDown Browser and a webcam for online exams. The webcam can be the type that's built into your computer or one that plugs in with a USB cable.
- Watch this brief video to get a basic understanding of LockDown Browser and the webcam feature: https://www.respondus.com/products/lockdown-browser/student-movie.shtml
- **Download Instructions:** Download and install LockDown Browser from the following link: https://download.respondus.com/lockdown/download.php?id=353467840
- **Once Installed**
  o Start LockDown Browser
  o Log into Blackboard Learn
  o Navigate to the test
- Note: You won't be able to access tests with a standard web browser. If this is tried, an error message will indicate that the test requires the use of LockDown Browser. Simply start LockDown Browser and navigate back to the exam to continue.
- **Guidelines**: When taking an online test, follow these guidelines:
  o Ensure you're in a location where you won't be interrupted
  o Turn off all other devices (e.g. tablets, phones, 2nd computers) and place them outside of your reach
  o Before starting the test, know how much time is available for it, and also that you've allotted sufficient time to complete it
  o Clear your desk or workspace of all external materials not permitted - phones and other devices (You are allowed paper notes and your textbook)
  o Remain at your computer for the duration of the test
  o If the computer, Wi-Fi, or location is different than what was used previously with the "Webcam Check" and "System & Network Check" in LockDown Browser, run checks again prior to the exam
  o To produce a good webcam video, do the following:
    ▪ Avoid wearing baseball caps or hats with brims
    ▪ Ensure your computer or device is on a firm surface (a desk or table). Do NOT have the computer on your lap, a bed, or other surface where the device (or you) are likely to move
    ▪ If using a built-in webcam, avoid readjusting screen tilt after the webcam setup is complete
    ▪ Take the exam in a well-lit room, but avoid backlighting (avoid window behind back)
  o Remember that LockDown Browser will prevent you from accessing other websites or applications; you will be unable to exit the test until all questions are completed and submitted
- **Getting Help:** Several resources are available if you encounter problems with LockDown Browser:
  - The Windows and Mac versions of LockDown Browser have a "Help Center" button on the toolbar. Use the "System & Network Check" to troubleshoot issues, and also run the "Webcam Check".
  - Respondus has a Knowledge Base available from support.respondus.com. Select the "Knowledge Base" link and then select "Respondus LockDown Browser" as the product. If your problem is with a webcam, select "Respondus Monitor" as your product
  - If you're still unable to resolve a technical issue with LockDown Browser, go to support.respondus.com and select "Submit a Ticket". Provide detailed information about your problem and what steps you took to resolve it.