Idaho State University | Informatics and Computer Science

Data Structures and Algorithms (3 credits)
Spring Semester 2020

CS 2235

Tuesdays and Thursdays

9:30 – 10:45am MT

ISU: CHE 309 (IF) / LIBR 16 (Pocatello)

Instructor: Dr. Leslie Kerby
Office: CAES 285
Email: kerblesl@isu.edu

T/A: Pepo Mena
Email: menapedr@isu.edu

Syllabus

COURSE DESCRIPTION

Implementation, usage, and design concerns of important data structures and their operations. Implementation and discussion of basic search and sorting algorithms. Discussion will include both O(n log n) and linear sorting algorithms. Incorporates aspects of time complexity and asymptotic analysis of algorithms. Students will be required to develop small to medium sized programs.

COURSE PREREQUISITES

CS 1181: Introduction to Programming

TEXTBOOK

Optional:
Data Structures & Algorithms in Java, Second Edition, Robert Lafore
Data Structures and Algorithms in Java, Sixth Edition, Goodrich, Tamassia, and Goldwasser

COURSE OBJECTIVES

Object-oriented Programming

- Define and use iterators and other operations on aggregates, including operations that take functions as arguments, in multiple programming languages, selecting the most natural idioms for each language. [U]

Basic Type System

- Give an example program that does not type-check in a particular language and yet would have no error if run. [F]
- Discuss the differences among generics, subtyping, and overloading. [F]
- Explain multiple benefits and limitations of static typing in writing, maintaining, and debugging software. [F]
- For multiple programming languages, identify program properties checked statically and program properties checked dynamically. [U]
- Use types and type-error messages to write and debug programs. [U]
- Define and use program pieces (such as functions, classes, methods) that use generic types, including for collections. [U]

Basic Analysis

- Explain what is meant by "best" ,"expected" ,and"worst" case behavior of analgorithm. [F]
- In the context of specific algorithms, identify the characteristics of data and/or other conditions or assumptions that lead to different behaviors. [A]
- State the formal definition of big O.[F]
- List and contrast standard complexity classes. [F]
- Give examples that illustrate time-space trade-offs of algorithms. [F]
- Determine informally the time and space complexity of simple algorithms. [U]
- Perform empirical studies to validate hypotheses about runtime stemming from mathematical analysis. Run algorithms on input of various sizes and compare performance. [A]

Fundamental Data Structures

- Describe common applications for each of the following data structures: stack, queue, priority queue, set, and map. [F]
- Compare and contrast the costs and benefits of dynamic and static data structure implementations. [A]
- Write programs that use each of the following data structures: arrays, records/structs, strings, linked lists, stacks, queues, sets, and maps. [U]
- Compare alternative implementations of data structures with respect to performance. [A]

Fundamental Data Structures and Algorithms

- Implement simple search algorithms and explain the differences in their time complexities. [A]
- Be able to implement common quadratic and O(N log N) sorting algorithms. [U]
- Solve problems using fundamental graph algorithms, including depth-first and breadth-first search. [U]
- Solve problems using graph algorithms, including single-source and all-pairs shortest paths, and at least one minimum spanning tree algorithm. [U]
- Trace and/or implement a string-matching algorithm. [U]
- Describe the implementation of hash tables, including collision avoidance and resolution. [F]
- Discuss the runtime and memory efficiency of principal algorithms for sorting, searching, and hashing. [F]
- Discuss factors other than computational efficiency that influence the choice of algorithms, such as programming time, maintainability, and the use of application-specific patterns in the input data. [F]
- Explain how tree balance affects the efficiency of various binary search tree operations. [F]
- Describe the heap property and the use of heaps as an implementation of priority queues. [F]

Algorithms and Design

- Implement a divide-and-conquer algorithm for solving a problem. [U]
- Apply the techniques of decomposition to break a program into smaller pieces. [U]
- Identify the data components and behaviors of multiple abstract data types. [U]
- Implement a coherent abstract data type, with loose coupling between components and behaviors. [U]
- Determine whether a recursive or iterative solution is most appropriate for a problem. [A]

Introductory Modeling and Simulation

- Create a simple, formal mathematical model of a real-world situation and use that model in a simulation. [U]
- Create a simple display of the results of a simulation. [U]
- Explain the concept of modeling and the use of abstraction that allows the use of a machine to solve a problem. [F]

ABSENCES

If for some reason you are absent from a class, it is your responsibility to contact your fellow students for what you missed in class. This includes homework assignments that were assigned, and additional content discussed.

ACADEMIC INTEGRITY

Academic integrity is expected at Idaho State University and the College of Business. All forms of academic dishonesty, including cheating and plagiarism, are strictly prohibited, the penalties for which range up to permanent expulsion from the university with "Expulsion for Academic Dishonesty" noted on the student's transcript. If you are unclear as to what constitutes academic dishonesty, you can get a copy of the College of Business Policy on Academic Integrity from the College of Business office in BA 202, or from the College of Business website at www.cob.isu.edu, or refer to the ISU Faculty/Staff Handbook policy on academic dishonesty at: http://www.isu.edu/policy/fs-handbook/part6/6_9/6_9a.html or http://www.isu.edu/library/research/ait/aitsitemap.html
Academic dishonesty includes, but is not limited to 1) cheating on examination, 2) plagiarism, or 3) collusion.

Definitions:

Cheating on an examination includes:

- Copying from another's paper, any means of communication with another during an examination, giving aid to or receiving aid from another during an examination;
- Using any material during an examination that is unauthorized by the proctor;
- Taking or attempting to take an examination for another student or allowing another student to take or attempt to take an examination for oneself.
- Using, obtaining, or attempting to obtain by any means the whole or any part of an unadministered examination.
- Talking to anyone other than the professor during an exam.

Plagiarism is the unacknowledged incorporation of another's work into work which the student offers for credit.

Collusion is the unauthorized collaboration of another in preparing work that a student offers for credit.

Other types of academic dishonesty include:

- Using other student's content from the labs or students' disk, etc.
- Performing any act designed to give unfair advantage to a student or the attempt to commit such acts is considered cheating.

Notes:

- The use of the source code of another person's program, even temporarily, is considered plagiarism.
- Copying material from a source without attributing (citing) the source.
- Allowing another person to use your source code, even temporarily, is considered collusion.

In this class, the specific exceptions given below are not considered scholastically dishonest acts:

- Discussion of the algorithm and general programming techniques used to solve a problem.
- Giving and receiving aid in debugging
- Discussion and comparison of program output (output only not code)

If the instructor finds an assignment that has been submitted as your own work and is not, it will be recorded as a 0. If you are found cheating on an exam, your exam will be taken and your result for the exam will be recorded as a 0.

EXPECTATIONS ON PROFESSIONALISM AND ATTITUDE

Be prepared for class. Read the assigned readings carefully before class, and spend some time thinking about what you've read. Write down questions about the reading as they occur to you.
Get to class on time.
Ask questions in class.
Stay alert and attentive in class.
Participate in class discussions in a courteous and positive manner.
Do not carry on "sidebar" conversations during instructor, student or guest presentations.
Laptops can be used to take notes, review notes or documents relevant to that day's class, or to quickly search for information to answer a question that has come up during discussion. If you are discovered to be reading email, participating in on-line chats, surfing the web or other non-class related activities using laptops or other accouterments during class, you will be warned and upon repeated occurrences requested to leave the class.
Spend enough time on assignments to ensure that you are actually learning the material and are able to turn in work of high quality. Assignments are an opportunity to learn, not something to just "get done."
Turn in assignments on time.

ACCOMMODATION

Students with disabilities are encouraged to discuss their needs with the instructor, preferably during the first week of class. All reasonable accommodations will be made to see that disabilities do not restrict a student's opportunity to learn. Students may also call the ADA and Disabilities Resource Center at 282-3599.

ASSIGNMENTS

Expect homework every week. Late homework will forfeit 10% of the grade if overdue less than one week, 25% of the grade if overdue more than one week and less than two weeks, and 100% after that.

SUCCESS

Success in this course depends heavily on your personal health and well-being. Recognize that stress is an expected part of the college experience, and it often can be compounded by unexpected setbacks or life changes outside the classroom. I encourage you to reframe challenges as an unavoidable pathway to success. Reflect on your role in taking care of yourself throughout the term, before the demands of exams and projects reach their peak. Please feel free to reach out to me about any difficulty you may be having that may impact your performance in this course. If you are experiencing stress in other areas of your campus life, I am happy to help you get in contact with other resources on campus that stand ready to assist you. In addition to your academic advisor, I strongly encourage you to contact the many other support services on campus that are available.

SERVICES

ISU Counseling and Testing Services (CATS) would like to remind all students who are enrolled in the current semester (part-time or full-time) they are eligible for free, confidential counseling services. CATS offers individual, group, and couples counseling, as well as Biofeedback Training. We also offers crisis intervention services Monday through Friday from 8-5.

Locations:

Pocatello: Graveley Hall, 3rd floor of the south side. To schedule an appointment call 208-282-2130 or just stop by. Walk-ins are welcome Monday-Friday from 8am - 4pm.

Idaho Falls: Bennion Student Union, 2nd floor, Room 223. Please call 208-282-7750 to schedule an appointment.

Meridian: Counseling services are available to Meridian students through the ISU Counseling Department. Please call 208-373-1719 to schedule an appointment.

COMFORTABLE LEARNING ENVIRONMENT

We are all committed to maintaining an inoffensive, non-threatening learning environment for every student. Class members (including the instructor) are thus to treat each other politely—both in word and deed. Offensive humor and aggressive personal advances are specifically forbidden. If you feel uncomfortable with a personal interaction in class, see your instructor for help in solving the problem.

GRADING

Course grading will be determined using the following breakdown:

Topic	Value
Homework	60%
Midterm	20%
Final exam	20%

Tentatively, grades will be earned based on a straight scale grading policy. The instructor reserves the right to change the grading scale, and assignment weighing. Such changes would be:

i) based on professional judgment
ii) applied across the board to all students
iii) in favor of the students

TENTATIVE COURSE SCHEDULE

Date	Topic
1/13 – 1/17	Overview and Java
1/20 – 1/24	
1/27 – 1/31	Fundamental Data Structures
2/3 – 2/7	Algorithm Analysis
2/10 – 2/14	Recursion
2/17 – 2/21	Stacks and Queues
2/24 – 2/28	List and Iterator ADTs
3/2 – 3/6	Trees
3/9 – 3/13	Priority Queues
3/16 – 3/20	Maps and Hash Tables
~~3/23 – 3/27~~	SPRING BREAK
3/30 – 4/3	Search Trees
4/6 – 4/10	Sorting and Selection
4/13 – 4/17	Text Processing
4/20 – 4/24	Graph Algorithms
4/27 – 5/1	Memory Management and B-trees (if time)
	Final Exam
5/4 – 5/8	