Comp 220: Data Structures, Fall 2017

*This syllabus is subject to change based on specific class needs, especially the schedule. Significant*

*deviations will be discussed in class.*

 Logistics

 Content

 Topics

 Sources

 Programming Environment

 Policies

 Assessment

 Assignments

 Workload

 Grading

 Schedule

 Monmouth College Services

Logistics

 Class meetings:

 Class: MWF 8:00 am - 8:50 AM in CFSB 323

 Lab: Th 2:00 pm - 3:50 PM in CFSB 309

 Instructor: Robert Utterback

 Office: CSB 342

 Phone: 309.457.2202

 Website: https://robertutterback.github.io

 Email: rutterback@monmouthcollege.edu

 Office hours: M 9-10 AM. Tu 3-4 PM. Th 11-noon. F 10-11 AM. By appointment (see the schedule on

my webpage).

 Website: https://robertutterback.github.io/courses/comp220/f17/

 Credits: 1 course credit

 Prerequisites: COMP160 and COMP161 with a C or better.

Content

Data structures continues the study of abstraction and programming through a focused study of data

structures, algorithms, and abstract data types. The primary focus of this course is the design and development

of algorithms and programs using data abstraction and information hiding via abstract data types *ADTs*. Data

abstraction is absolutely fundamental to good programming practice and the management of large scale

problems and data sets. This course is designed to round out a student’s understanding of basic computer

science concepts while strengthening the program design and development skills through continued use of a

test-driven design methodology.

/

It is imperative that programmers and program designers be able to determine which of the many solutions available is the best for their specific task. Throughout the course, students will focus on to making solid quantitative and qualitative judgments about program efficiency and overall program design choices. Gone are the days when “it works” is a good enough assessment for the quality of a program.

Students will explore the ideas and concepts brought up in class and homework assignments during the weekly lab session. In addition to hands on exercises, lab sessions will be used to explore current, relevant research in computer science.

Topics

We will focus on chapter 5 and beyond from the text. This includes, but is not limited to:

 Review of c++ fundamentals (chapters 1–4, comp161 notes)

 Review of algorithm design, analysis through sorting algorithms (chapter 10 \& notes)

 Standard container types (chapter 5)

 Classes and adt design (chapter 6)

 Recursive strategies (parts of chapters 7–9)

 Pointer and array logic (chapter 11)

 Dynamic memory management (chapter 12)

 Efficient implementation of container types such as vectors, lists, stacks, queues, trees, maps, sets, and graphs. (chapters 13–17)

Sources

The course textbook will be:

 Roberts, Eric S. *Programming Abstractions in C++*. Pearson. 2014. ISBN-13:978-0133454840

Other sources will be posted on this webpage as needed.

Programming Environment

This course utilizes the Code::Blocks ide and the GNU compiler for C++ development.

All programs written in this course are required to compile and run on a server running Ubuntu 16.04.3. All students will have access to the departmental server and thereby the above development tools. While development is not required in this environment, *failure to properly port a program to the required environment could result in your program not compiling correctly at the time of grading*. All software for this course is available free of charge and can be found on the web if students wish to install it on their personal machines.

Policies

 **Late assignments**: In general, late assignments will *not* be accepted. Exceptions may be made only for situations beyond your control. If you feel your reason is justified, schedule a meeting with the instructor to plead your case.

 **Academic dishonesty**: Monmouth College’s official policy on academic dishonesty can be found here. **You** are responsible for reading and complying with that policy.

*In this course, any violation of the academic honesty policy will have varying consequences depending on the severity of the infraction as judged by the instructor. Minimally, a violation will result in an “F” or 0 points on the assignment in question. Additionally, the student’s course grade may be lowered by one letter*  /

*grade. In severe cases, the student will be assigned a course grade of “F” and dismissed from the class. All cases of academic dishonesty will be reported to the Associate Dean who may decide to recommend further action to the Admissions and Academic Status Committee, including suspension or dismissal. It is assumed that students will educate themselves regarding what is considered to be academic dishonesty, so excuses or claims of ignorance will not mitigate the consequences of any violations*

 **Collaboration**: We encourage you to make use of the resources available to you – it is fine to seek help from a friend, tutor, instructor, internet, etc. However, *copying of answers and any act worthy of the label of* *“cheating” is never permissible*! It is understandable that when you work with a partner or a group that the resultant product is often extremely similar. This is acceptable but be prepared to be asked to defend your collaborations to the instructor. *You should always be able to reproduce an answer on your own, and if you*  *cannot you likely* ***do not really know the material***.

One way to collaborate effectively is to avoid taking careful notes during a collaboration session. Discuss the material and sketch out possible solutions on a whiteboard. When you have finished, take a break and then write up your solutions without any help from notes or pictures from the study session. This not only helps avoid violations of academic dishonesty, it also improves your retention of the material!

When assignments are meant to be done in groups, you will be directed to turn in one set of solutions per group. Otherwise, each student must turn in an assignment representing their own work.

 **Electronic devices**: Do not use your phone in class. Keep it on silent or leave it at home. Any computer or tablet usage should be related to the course. Other usage is rude and distracting to others.

 **General expectations**: In short, I expect you to be respectful of others and take responsibility for your own learning. You are here to learn, so work hard and be professional.

 Just attending class is not sufficient to truly learn the material. Read the text, use the resources available at Monmouth College, and go beyond the material.

 If you miss class, you are responsible for everything covered on that day. College is, in some sense, your job. Take pride in creating quality work. Staple your assignments, label problems, and present your answers neatly and orderly.

 Your job is to convince me that you have learned the material – show your work! Even if you do not know a particular answer, guide me through your thought process.

Assessment

Assignments

The course workload is as follows:

|  |  |
| --- | --- |
| **Category** | **Number Of Assignments** |
| Labs | 10 |
| Homework | 8–10 |
| Projects | 2 |
| Exams | 7 |

/

Homework assignments will always either precede a lab to prepare for it or follow a lab to complete it. Students are also encouraged to look at the textbook’s review questions, since the solutions are available online. There will be no dedicated midterm or final exam, but 7 exams spaced throughout the semester. Each exam will focus primarily, but not necessarily exclusively, on the material covered since the previous exam. The final exam will include a small number of cumulative questions, and I reserve the right to include at most one cumulative question on each of the other exams.

Workload

The weekly workload for this course will vary by student but on average should be about 13 hours per week. The follow tables provides a rough estimate of the distribution of this time over different course components for a 16 week semester.

|  |  |  |
| --- | --- | --- |
| **Category** | **Total Time** | **Time/Week (Hours)** |
| Lectures+Labs |  | 4 |
| Homework | 48 | 3 |
| Exam Study | 16 | 1 |
| Projects | 48 | 3 |
| Reading+Unstructured Study |  | 2 |

13

Grading

Lab and homework assignments are graded on a simple 3 point scale. Grades are marked with, in decreasing order, a check-plus, check, or check-minus. Your final grade for these two assignment categories is then based off the respective averages and determined by the following chart. Notice this chart lists the minimum average needed to achieve a particular letter grade.

|  |  |
| --- | --- |
| **Assignment Avg. (min)** | **Letter Grade** |
| 2.8 | A |
| 2.75 | A- |
| 2.5 | B+ |
| 2.25 | B |
| 2 | B- |
| 1.75 | C+ |
| 1.5 | C |
| 1 | C- |
| 0.75 | D |
| 0.5 | F |

Your final grade is based on a weighted average of particular assignment categories. You can estimate your current grade based on your scores and these weights. You may always visit the instructor *outside of class* to /

discuss your current standing.

|  |  |
| --- | --- |
| **category** | **weight** |
| exams | 45% |
| projects | 25% |
| homework | 12.5% |
| labs | 12.5% |
| participation | 5% |

This courses uses a standard grading scale. Assignments and final grades will not be curved except in rare

cases when its deemed necessary by the instructor. Percentage grades translate to letter grades as follows:

|  |  |
| --- | --- |
| **Score** | **Grade** |
| 94–100 | A |
| 90–93 | A- |
| 88–89 | B+ |
| 82–87 | B |
| 80–81 | B- |
| 78–79 | C+ |
| 72–77 | C |
| 70–71 | C- |
| 68–69 | D+ |
| 62–67 | D |
| 60–61 | D- |
| 0–59 | F |

You are always welcome to challenge a grade that you feel is unfair or calculated incorrectly. Mistakes made in

your favor will never be corrected to lower your grade. Mistakes made not in your favor will be corrected.

*Basically, after the initial grading your score can only go up as the result of a challenge*.

Schedule   
The following **tentative** calendar should give you a feel for how work is distributed throughout the semester.

Assignments and events are listed in the week they are due or when they occur. *This calendar is subject to*

*change based on the circumstances of the course*.

|  |  |  |  |
| --- | --- | --- | --- |
| **date** | **topic** | **Assignment** |  |
| Fri 12/08 6:30 |  | (Final) Exam 7 (Solutions) |  |
| PM |
| Wed 12/13 |  | “HW” 8 due | / |

|  |  |  |
| --- | --- | --- |
| **date** | **topic** | **Assignment** |
| Wed 08/23 | Intro | Using VNC, Code::Blocks, COMP161 |
| BigO Notes |
| Thu 08/24 | Logarithms (notes) | Lab/HW 1 assigned |
| Fri 08/25 | Review C++ Fundamentals and Big | COMP161 C++ Notes I, COMP161 C++ |
| O | Notes II |
| Mon 08/28 | Analyzing Simple Recursion (notes) | Read Ch. 7 |
| Wed 08/30 | Palindromes and Binary Search |  |
| Thu 08/31 | Recursion to Iteration | HW 1 due, Lab/HW 2 AND HW 3 |
| Fri 09/01 | Review |  |
| Mon 09/04 | Exam 1 |  |
| Wed 09/06 | Structural Recursion and Iteration |  |
| (notes) |
| Thu 09/07 | Recursion for intToStr | HW 2 AND 3 due, Lab 3 |
| Fri 09/08 | Analyzing Sorting | Read Ch. 10 |
| Mon 09/11 | Divide and Conquer Sorting |  |
| (Mergesort) |
| Wed 09/13 | Quicksort |  |
| Thu 09/14 | Tracing Sorting and Searching | Lab/HW 4 (Solutions) |

Fri 09/15

|  |  |  |
| --- | --- | --- |
| Mon 09/18 | Collections: Stacks and Queues | Read 5.1-5.3 |
| Wed 09/20 | Exam 2 |  |
| Thu 09/21 | Using Stacks and Queues | Lab/HW 5 |
| Fri 09/22 | Collections: Maps and Sets | Read 5.4-5.5 |
| Mon 09/25 | Traversing Collections (notes) | Read 5.6 |
| Wed 09/27 | Designing Classes | Read 6.1 |
| Thu 09/28 | Map Inversion | Lab/HW 6 |

Fri 09/29

Mon 10/02

|  |  |  |  |
| --- | --- | --- | --- |
| Wed 10/04 | Exam 3 | Solutions |  |
| Thu 10/05 | Stacks Using Vectors | Lab/HW 7 |  |
| Fri 10/06 | Overloading Operators | Read 6.1 |  |
| Fri 12/08 6:30 |  | (Final) Exam 7 (Solutions) |  |
| PM |
| Wed 12/13 |  | “HW” 8 due | / |

|  |  |  |
| --- | --- | --- |
| **date** | **topic** | **Assignment** |
| Mon 10/09 |  | Exam 4 (take-home) |
| (10/11–10/15) | (Fall Break) | (Fall Break) |
| Mon 10/16 | Exceptions and Exception handling |  |
| (notes) |
| Wed 10/18 | Handling Input: Tokenizing | Read 6.4 |
| Thu 10/19 | Reverse Polish Notation Calculator | Project 1 out |
| Fri 10/20 | Pointers |  |
| Mon 10/23 |  | Read 11.1-11.2 |
| Wed 10/25 | Arrays and Pointer Arithmetic | Read 11.3-11.4 |
| Thu 10/26 | Free Lab for Project 1 |  |
| Fri 10/27 | C-style Strings |  |
| Mon 10/30 | Dynamic Memory, Linked Lists | Read 12.1-12.2 |
| Wed 11/01 | Traversing Linked Lists, Arrays vs. |  |
| Linked Lists |
| Thu 11/02 | Array comparison | Lab 8 |
| Fri 11/03 | Freeing Memory, Destructors | Exam 5 (take-home), Read 12.3 |
| Mon 11/06 | DoubleStack | Read 12.4-12.5 |

Wed 11/08

|  |  |  |  |
| --- | --- | --- | --- |
| Thu 11/09 | Better Arrays and Vectors | Lab 9/HW 7 |  |
| Fri 11/10 | Copying Objects | Read 12.7 |  |
| Mon 11/13 | Dynamic-array Stack Analysis, | Read 12.8, 14.1-14.2 |  |
| Templates |
| Wed 11/15 | Implementing Queues | Read 14.3 |  |
| Thu 11/16 |  | Project 2 out, Exam 6 out |  |
| Fri 11/17 | Implementing Vectors | Labp2 due, Read 14.4-14.5 |  |
| Mon 11/20 | Debugging in CodeBlocks; Maps | Exam 6 due 11/21, Read 15.1 |  |
| with Vectors |
| (11/22–11/26) | (Thanksgiving Break) | (Thanksgiving Break) |  |
| Mon 11/27 | Hashing | Read 15.2-15.4 |  |
| Wed 11/29 | Hash Table Load Factor; Intro to Tree | Read 16.1-16.2 |  |
| Thu 11/30 | Free lab for Project 2 |  |  |
| Fri 12/08 6:30 |  | (Final) Exam 7 (Solutions) |  |
| PM |
| Wed 12/13 |  | “HW” 8 due | / |

|  |  |  |
| --- | --- | --- |
| **date** | **topic** | **Assignment** |
| Fri 12/01 | Implementing BSTs | Project 2 Due |
| Mon 12/04 | Tree Traversals, Balanced Trees | Read 16.3, “HW” 8 out |
| Wed 12/06 | Tree Rotations, Review |  |
| Fri 12/08 6:30 |  | (Final) Exam 7 (Solutions) |
| PM |
| Wed 12/13 |  | “HW” 8 due |

Monmouth College Services

 The **Teaching and Learning Center** offers FREE resources to assist Monmouth College students with their academic success. Programs include Supplemental Instruction for difficult classes, Drop-In and appointment tutoring, and individual Academic Coaching. The TLC is here to help students excel academically. TLC services are not just for struggling students, but can assist all students to get better grades, practice stronger study skills, and manage time. The TLC is located on the 2nd floor of Poling Hall.

 **Disability Support Services**: If you have a disability or had academic accommodations in high school or another college, you may be eligible for academic accommodations at Monmouth College under the Americans with Disabilities Act (ADA). Monmouth College is committed to equal educational access.

Students with disabilities can apply for accommodations at the Teaching and Learning Center.

 If you have not been diagnosed with a learning disability but believe that you would benefit from disability screening, please contact Amy Hofmeister in Counseling Services.

 The instructor will be notified of students with accommodations; however, it is the student’s   
 responsibility to activate her/his accommodations. Please meet with the instructor ASAP if this applies to you!

/