Syllabus
CS 23001 Computer Science II: Data Structures & Abstraction
Spring 2022
Department of Computer Science
Kent State University

Instructor: Dr. Jonathan I. Maletic and Office Hours

Course Time: MW 2:15pm - 3:30pm
Course Location: 110 Smith Hall

Course Description: Advanced computer programming design, and development with a primary focus on data structures and abstraction using an object oriented programming language.

Prerequisite: CS 13011 Computer Science I-A and CS 13012 Computer Science I-B both with C (2.0) or better
Pre/Co requisite: CS 23022 Discrete Structures for Computer Science (need to have taken or be taking same term)

Text and Reference Books (optional, not required):

- *Data Structures and Other Objects Using C++*, Main and Savitch, Addison Wesley
- *C++ Plus Data Structures*, Dale, Free online access via KSU library/Safari
- *The C++ Programming Language*, Stroustrup, B., Addison Wesley
- *Practical Debugging in C++*, Ford & Teorey, Prentice Hall

Course website: [www.cs.kent.edu/~jmaletic/CS23001/](http://www.cs.kent.edu/~jmaletic/CS23001/)

Course Objectives:

- Continue developing a disciplined approach to problem solving methods and algorithm development.
- Provide a clear understanding of the concepts of abstract data types.
• To teach a number of the basic algorithms and data structures used in computer science.
• To teach the concepts of object oriented programming.
• To provide a foundation for further studies in Computer Science.
• On completion of this course, students must have a basic understanding of the concepts of abstract data types and object oriented programming methods. Data structures such as lists, stacks, queues, strings, and trees must be understood. The student will have working knowledge of the concepts of classes and objects, operator overloading, constructors, destructors, and generics. The concepts of dynamic data structures and recursion must be well understood.

**Learning Objectives:** Students will be able to apply and compare the basic abstract data structures used in computer science. These include strings, sets, stacks, queues, lists, and binary trees. Students will understand and demonstrate their ability to construct abstract data types and solve problems using an object oriented programming language. This will include demonstrated understanding of recursion, dynamic memory management, generics, and operator overloading.

**Course Organization and Grading:**
Lecture and Lab attendance is a critical component of the course and is reflected in the final grade. The class participation grade will be determined in part by attendance along with in class quizzes or assignments. Additionally, credit may be given participating in online discussion. These assignments will be completed during the class period and handed in during class. No makeups for inclass assignments will be given. The lab participation grade will also be derived from attendance and completing assignments during lab. Grades will be available via your Flashline account on blackboard.

<table>
<thead>
<tr>
<th>%</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>25%</td>
<td>Midterm Exam - Week of Feb. 28th in Lab</td>
</tr>
<tr>
<td>35%</td>
<td>Final Exam - Monday May 9, 12:45 - 3:00pm (MSB 162, 139, 243)</td>
</tr>
<tr>
<td>25%</td>
<td>Projects (4 projects, a part due each week)</td>
</tr>
<tr>
<td>5%</td>
<td>Lecture participation (attendance/inclass and online discussion)</td>
</tr>
<tr>
<td>10%</td>
<td>Laboratory (lab attendance and lab assignments)</td>
</tr>
</tbody>
</table>

**Final Grading Scale:**

<table>
<thead>
<tr>
<th>Grade</th>
<th>GPA</th>
</tr>
</thead>
<tbody>
<tr>
<td>F</td>
<td>0.00</td>
</tr>
<tr>
<td>D</td>
<td>1.00</td>
</tr>
<tr>
<td>D+</td>
<td>1.30</td>
</tr>
<tr>
<td>C-</td>
<td>1.70</td>
</tr>
<tr>
<td>C</td>
<td>2.00</td>
</tr>
<tr>
<td>C+</td>
<td>2.30</td>
</tr>
<tr>
<td>B-</td>
<td>2.70</td>
</tr>
<tr>
<td>B</td>
<td>3.00</td>
</tr>
<tr>
<td>B+</td>
<td>3.30</td>
</tr>
<tr>
<td>A-</td>
<td>3.70</td>
</tr>
<tr>
<td>A</td>
<td>4.00</td>
</tr>
</tbody>
</table>
Lab:
There is a laboratory associated with lecture. The lab is to complement lecture and support the application of materials learned. There will be a lab instructor; attendance and participation is required. The lab will consist in part of short lectures by the lab instructor and hands on exercises that will assist in the learning process. For each lab meeting there will be a score given and is based on attendance and successful completion of lab exercises.

Projects:
There will be at four substantial projects given over the term. These can be completed partially during the lab time however, the bulk of the work will be completed outside of lab/lecture time. The requirement for these assignments will be posted on the course web page. The projects will be broken down into three to four parts. Each part will have its own due date and count as a percentage of the total project grade. Due dates for each part of the project will be posted on the course website. The general topics of the programs will be:

- Abstract Data Types & Classes
- Dynamic Memory and Containers
- Linked lists and templates
- Dynamic Data Structures (Complex Pointers)

Other Notes:

- Lecture is the student's responsibility, if class is missed; it is in the students best interests to get the notes from a fellow student. The instructor does not have slides or lecture notes to hand out.
- There will be no make up exams.
- There will be no make ups for in class assignments or quizzes.
- Please mute the sound on phones, lab tops, and other electronic devices.
- Any modifications to the syllabus will be made on this page and noted.
- This course was previously numbered CS 33001. It was changed to CS 23001 in Fall 2012. There was no change in content in the course due to this numbering change.

University Requirements and Announcements:
• Registration Requirement: The official registration deadline for this course can be found on the Registrars calendar. University policy requires all students to be officially registered in each class they are attending. Students who are not officially registered for a course by published deadlines should not be attending classes and will not receive credit or a grade for the course. Each student must confirm enrollment by checking his/her class schedule (using Student Tools in FlashFast) prior to the deadline indicated. Registration errors must be corrected prior to the deadline. Every class has its own schedule of deadlines and considerations. To view the add/drop schedule and other important dates for this class, go to the Students Tools and Courses tab in FlashLine and choose either View or Print Student Schedule. To see the deadlines for this course, click on the CRN or choose the Drop or Add a Course link and click on the green clock next to the course under Registration Deadlines.

• The Student Survey of Instruction (SSI) are administered online via Flash Survey.

• University policy 3342-3-01.3 requires that students with disabilities be provided reasonable accommodations to ensure their equal access to course content. If you have a documented disability and require accommodations, please contact the instructor at the beginning of the semester to make arrangements for necessary classroom adjustments. Please note, you must first verify your eligibility for these through Student Accessibility Services (contact 330-672-3391 or visit www.kent.edu/sas for more information on registration procedures).

• University policy 3342-3-01.8 deals with the problem of academic dishonesty, cheating, and plagiarism. None of these will be tolerated in this class. The sanctions provided in this policy will be used to deal with any violations. If you have any questions, please read the policy or contact the instructor. For the University’s complete policy on procedure on cheating and plagiarism go to: www.kent.edu/policyreg/index.cfm and search for policy 3-01.8.

Course Content:
Abstract Data Types (ADTs) and Object Oriented Concepts:

• Definition of ADTs
• Encapsulation and information hiding
• Classes, methods, constructors, and destructors
• Information hiding: Public, private, (and protected)
• Operator overloading and polymorphism
• Generics (templates)
• Inheritance, polymorphism, and virtual functions (dynamic variable binding)

Dynamic Memory Structures:
• Allocation and de-allocation of memory (new, delete)
• Dynamic Arrays
• Pointers, Linked Lists (insertion, deletion, etc.)

Abstract Data Structures & Algorithms:
• Array, multi-dimensional arrays records, files, strings
• Lists, stacks, and queues, sets, bags, vectors
• Containers and iterators
• Infix, prefix, and postfix notations and conversion algorithms
• Binary trees, binary search trees
• Recursion: Design and implementation of recursive functions
• Hashing and priority queues
• Brief introduction to graphs & associated algorithms

Additional Topics:
• Multi-file programs, make
• Testing and debugging techniques
• Exception handling

URL:http://www.cs.kent.edu/~jmaletic/CS23001/syllabus.html
Last update: Fri Jan 7 14:56:32 2022 EST