Managing Game Data  
ITP 365x (3 Units)

| **Objective** | This course is an overview of data structures commonly used in games. By the conclusion of the course, students will have:  
1. Familiarity with several fundamental game data structures.  
2. An understanding of big-O notation and time complexity.  
3. Knowledge of several basic and advanced sorting algorithms.  
4. Learn advanced object-oriented paradigms. |
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Prerequisites</strong></td>
<td>ITP 165x or equivalent experience</td>
</tr>
<tr>
<td><strong>Language</strong></td>
<td>This course is taught in C++, and at many points relies on knowledge of dynamic memory allocation. Because of this, the student must have taken either ITP 165x or have equivalent experience with C++. Knowledge of Java, Python, or another high level language other than C++ will likely not be sufficient.</td>
</tr>
<tr>
<td><strong>Instructor</strong></td>
<td>Sanjay Madhav (<a href="mailto:madhav@usc.edu">madhav@usc.edu</a>)</td>
</tr>
<tr>
<td><strong>Lecture/Lab</strong></td>
<td>3 hours per week.</td>
</tr>
<tr>
<td><strong>Course Structure</strong></td>
<td>Every week, we will cover one specific data structure in lecture. Some data structures are a bit more complex, so may be covered over multiple weeks. Most weeks, there will be a lab assignment which is due the following week. There are a total of 12 lab assignments, and each one covers a different aspect of game data structures. As there are many assignments, it is important to stay on top of them and not fall behind.</td>
</tr>
</tbody>
</table>
| **Grading** | The course is graded with the following weights:  
Weekly Assignments (5% each) | 60%  
Midterm Exam | 20%  
Final Exam | 20%  
TOTAL POSSIBLE | 100% |
Grading Scale

Letter grades will be assigned according to the following scale:

<table>
<thead>
<tr>
<th>Percentage</th>
<th>Grade</th>
</tr>
</thead>
<tbody>
<tr>
<td>93%+</td>
<td>A</td>
</tr>
<tr>
<td>90-92%</td>
<td>A-</td>
</tr>
<tr>
<td>87-89%</td>
<td>B+</td>
</tr>
<tr>
<td>83-86%</td>
<td>B</td>
</tr>
<tr>
<td>80-82%</td>
<td>B-</td>
</tr>
<tr>
<td>77-79%</td>
<td>C+</td>
</tr>
<tr>
<td>73-76%</td>
<td>C</td>
</tr>
<tr>
<td>70-72%</td>
<td>C-</td>
</tr>
<tr>
<td>69</td>
<td>D+</td>
</tr>
<tr>
<td>67-68</td>
<td>D</td>
</tr>
<tr>
<td>66</td>
<td>D-</td>
</tr>
<tr>
<td>65 and below</td>
<td>F</td>
</tr>
</tbody>
</table>

Half percentage points will be rounded up to the next whole percentage. So for instance, 89.5% is an A-, but 89.4% is a B+.

There is no curving. Students will receive the grade they earn. Extra credit is generally not offered.

Policies

Make-up policy for exams: To make up for a missed exam, the student must provide a satisfactory reason (as determined by the instructor) along with proper documentation. Make-up exams are only allowed under extraordinary circumstances.

Late Assignments: Late assignments will be accepted with a 10% penalty per day late. So if, for example, an assignment is turned in two days late, the maximum possible grade is 80%.

Before logging off a computer, students must ensure that they have emailed or saved projects created during the class or lab session. Any work saved to the computer will be erased after restarting the computer.

ITP is not responsible for any work lost.

ITP offers Open Lab use for all students enrolled in ITP classes. These open labs are held beginning the second week of classes through the last week of classes. Please contact your instructor for specific times and days for the current semester.
**Academic Integrity**

USC seeks to maintain an optimal learning environment. General principles of academic honesty include the concept of respect for the intellectual property of others, the expectation that individual work will be submitted unless otherwise allowed by an instructor, and the obligations both to protect one’s own academic work from misuse by others as well as to avoid using another’s work as one’s own. All students are expected to understand and abide by these principles. *SCampus*, the Student Guidebook, ([www.usc.edu/scampus](http://www.usc.edu/scampus) or [http://scampus.usc.edu](http://scampus.usc.edu)) contains the University Student Conduct Code (see University Governance, Section 11.00), while the recommended sanctions are located in Appendix A.

Students will be referred to the Office of Student Judicial Affairs and Community Standards for further review, should there be any suspicion of academic dishonesty. The Review process can be found at: [http://www.usc.edu/student-affairs/SJACS/](http://www.usc.edu/student-affairs/SJACS/).

Information on intellectual property at USC is available at: [http://usc.edu/academe/acsen/issues/ipr/index.html](http://usc.edu/academe/acsen/issues/ipr/index.html).

---

**Students with Disabilities**

Any student requesting academic accommodations based on a disability is required to register with Disability Services and Programs (DSP) each semester. A letter of verification for approved accommodations can be obtained from DSP. Please be sure the letter is delivered to me (or to TA) as early in the semester as possible. DSP is located in STU 301 and is open 8:30 a.m.–5:00 p.m., Monday through Friday.

Website and contact information for DSP: [http://sait.usc.edu/academicsupport/centerprograms/dsp/home_index.html](http://sait.usc.edu/academicsupport/centerprograms/dsp/home_index.html), (213) 740-0776 (Phone), (213) 740-6948 (TDD only), (213) 740-8216 (FAX) [ability@usc.edu](mailto:ability@usc.edu).

---

**Emergency Preparedness**

In case of a declared emergency if travel to campus is not feasible, USC executive leadership will announce an electronic way for instructors to teach students in their residence halls or homes using a combination of Blackboard, teleconferencing, and other technologies.

Please activate your course in Blackboard with access to the course syllabus. Whether or not you use Blackboard regularly, these preparations will be crucial in an emergency. USC's Blackboard learning management system and support information is available at [blackboard.usc.edu](http://blackboard.usc.edu).
# Course Outline

**Week 1** – Introduction to Game Data Structures  
- Templates  
- Review of Classes  
- Big-O Notation Basics  
**Reading: Sherrod: Chapter 1**  
**Assignment 1**: Vectors

**Week 2** – Arrays and Vectors  
- Dynamic Memory Review  
- Dynamic Arrays  
- Bit Arrays  
**Reading: Sherrod: Chapter 2**  
**Assignment 2**: Fibonacci Sequence (Recursive/Iterative)

**Week 3** – Recursion  
- Basic recursive algorithms  
- Recursive pitfalls  
- Factorials  
**Reading: Sherrod: Chapter 3**  
**Assignment 3**: Bubble vs. Insertion Speed Comparison

**Week 4** – Sorting Basics  
- Bubble Sort  
- Selection Sort  
- Insertion Sort  
**Reading: Sherrod: Chapter 4 (pp. 91-109)**  
**Assignment 4**: Basic Linked List

**Week 5** – Linked Lists  
- Properties of linked lists  
- Singly-linked lists  
- Doubly-Linked Lists  
**Reading: Sherrod: Chapter 5**  
**Assignment 5**: An OOP Dilemma

**Week 6** – Object-Oriented Programming, Part 1  
- Inheritance  
- Polymorphism  
- Is-a vs. Has-a  
**Reading: Posted on Blackboard**  
**Assignment 6**: DMV Line Tracker

**Week 7** – Midterm Exam

**Week 8** – Stacks and Queues  
- Stacks  
- Queues  
**Reading: Sherrod: Chapter 6**  
**Assignment 6**: DMV Line Tracker
**Week 9** – Hash Tables
- Hash Function
- Setting up buckets
- Array of linked lists implementation

*Reading: Sherrod: Chapter 7*

*Assignment 7: Hashing into Buckets*

**Week 10** – Advanced Sorting
- Merge Sort
- Quick Sort

*Reading: Sherrod: Chapter 4 (pp. 109-155); Chapter 8 (pp. 265-275)*

*Assignment 8: Quick Sort*

**Week 11** – Trees
- Basic Binary Trees
- Add/Removal
- Traversal of Trees

*Reading: Sherrod: Chapter 9 (pp. 285-315)*

*Assignment 9: BFS and DFS*

**Week 12** – Heaps
- Basic Heaps
- Heap Sort
- Priority Queues

*Reading: Sherrod: Chapter 10*

*Assignment 10: Heap Sort*

**Week 13** – Graphs
- Graph Basics
- Data Representation
- Basic Graph Search

*Reading: Sherrod: Chapter 10 (pp. 351-376)*

*Assignment 11: Adjacency Lists*

**Week 14** – Object-Oriented Programming, Part 2
- Abstract base classes
- Choosing the correct paradigm
- Design Patterns basics

*Reading: Posted on Blackboard*

*Assignment 12: Basic Design Patterns*

**Week 15** – Advanced STL
- map, set
- unordered_map, unordered_set
- algorithm

*Reading: Sherrod: Chapter 12*

**Week 16** – Final Exam