

# **CSE 373**

**SEPTEMBER 27 – COURSE  
INTRODUCTIONS;**

**ADTS; STACKS AND QUEUES**

# **WELCOME!**

- **Administrative Minutiae**
- **Course Objectives**
- **Review of Stacks and Queues**
- **Abstract Data Types (ADT)**

# **WAITLIST/OVERLOAD**

- **Because many students move around in the first week, the overload form will come out on Monday**
- **Please plan to attend lectures until then**

# COURSE INFO

- **Evan McCarty ([ejmcc@uw.edu](mailto:ejmcc@uw.edu))**
- **Office hours (CSE 214)**
  - Mondays and Fridays: 3:30 – 5:00
  - By appointment or over email

# **PIAZZA**

- **Primary method of course information**
- **Great place to answer questions**
- **Feedback to other students/Tas**
- **Preferred because it allows all students to see answers when there are common problems**

# **COURSE STAFF**

- **TAs will be posted on the course website, along with their office hours**
- **TAs for sections will be posted this evening**
- **Office hours will start on Monday**

# **HOMEWORK**

- **Homework in this course will be broken down into 3 projects and 2 written assignments**
- **40% of your total grade**

# PROJECTS

- **Partners allowed**
  - If you complete the projects as a partner, there will also be a small written portion of the project
  - Partners are highly recommended



# PROJECTS

- **Multiple parts**
  - Even though there are only 3 projects, they will span multiple weeks and have deliverables due each week

# **WRITTEN ASSIGNMENTS**

- **Written assignments must be completed alone**
- **One week from assignment to due date**

# **LATE POLICY**

- **Each student will have 3 late days**
- **If used by a team, both students must have a late day remaining**
- **Max of 2 late days per assignment**
- **No benefit for left over late days**

# **LATE POLICY**

- **15% per day late if the student has no late days left**
- **Late days automatically deducted, no choosing 15%**
- **0% given for assignments turned in more than 3 days after due date**
- **All assignments due at 11:30 pm, I am fairly lenient with a minute or two**

# **HOMEWORK**

- **Regrade requests**
  - Catalyst survey will be up this week and put onto the course webpage
  - Must be completed before midterm (for HW in first half) or before final (for HW in second half)

# **HOMEWORK**

- **Academic honesty**
  - High level discussion
  - Fully understand submission
- **Reasonable effort and office hours**

# **LECTURES**

- **Lecture slides will be posted online after class**
- **Questions are strongly encouraged**
- **All material fair game for exams**
- **Weiss textbook**

# **LECTURE STYLE**

- **Slides are largely for information**
- **Not necessarily enough for understanding**
- **Document projector**
- **Peer instruction**
- **Ask questions!**



# LECTURE STYLE

- **Slides are largely for information**
- **Not necessarily enough for understanding**
- **Document projector**
- **Peer instruction**
- **Ask questions! Point out mistakes!**

# SECTIONS

- **Conducted by TAs**
- **Practice problems**
  - Analysis and Implementation
- **Supplementary instruction**
- **Section tomorrow**

# EXAMS

- **Midterm exam (25%)**
  - 2:30 – 3:20; Friday, November 3
- **Final Exam (35%)**
  - 2:30-4:20; Tuesday, December 12
- **Exam review in lecture before exams**

# PROJECT 0

- **Ungraded**
- **Install Eclipse**
  - Required for this course, project files will be in eclipse project format
  - JGrasp will not handle project packages
- **Ensure that you have Java 8**
  - (Java 9 has just come out)
- **Instructions out tonight**

# **DATA STRUCTURES AND ALGORITHMS**

# **DATA STRUCTURES AND ALGORITHMS**

- **Understand and recognize behavior of key data structures**
- **Understand and solve common data structure problems**
- **Analyze operations and algorithms**
- **Implement data structures and understand design trade-offs**

# CSE 143

- **Object-oriented Programming**
  - Classes and Inheritance
  - Methods, variables and conditions
  - Loops and recursion
  - Linked lists and simple trees
  - Basic Sorting and Searching
  - Concepts of Analysis  $O(n)$  v  $O(n^2)$
  - Client v. Implementer

# **CSE 373**

- **Design decisions**
- **Critical thinking**
- **Implementations**
- **Debugging and Testing**
- **Abstract Data Types**
- **Code-base development**



# ABSTRACTION

- **Software engineering  
v. Computer Science**
- **Applicable across languages and  
implementations**
- **Behavior focus**
  - How can you recognize an ADT?

# STACK?

- **What is a stack?**

# STACK?

- **What is a stack?**
  - Outside of CS?

# STACK?

- **What is a stack?**
  - Outside of CS?
  - From 143?

# DEFINITIONS

- **Abstract Data Type (ADT)**
  - Operations and expected behavior
- **Data Structure**
  - Specific organization of data
  - Can be analyzed
- **Implementation**
  - Language specific application

# DESIGN DECISIONS

- **Between an ADT and its implementation, there are design decisions**
- **Constraints of the problem**
  - Memory v. Speed
  - One function v. another
  - Generality v. Specificity

# DESIGN DECISIONS

- **Linked List v Array**
  - From 143, what are some advantages/disadvantages?
  - Why?

# DESIGN DECISIONS

- **Linked List v Array**
  - Overhead
  - Memory use
  - Adding to middle
  - Traversal
  - Insertion



# DESIGN DECISIONS

- **Shopping list?**

# DESIGN DECISIONS

- **Shopping list?**
  - What sorts of behavior do shoppers exhibit?
  - What constraints are there on a shopper?
  - What improvements would make a better shopping list?

# DESIGN DECISIONS

- **Shopping list?**
  - Behavior

# DESIGN DECISIONS

- **Shopping list?**
  - Constraints

# DESIGN DECISIONS

- **Shopping list?**
  - Improvements