

**CS367 Introduction to Data Structures  
University of Wisconsin-Madison  
Summer 2014**

**Course website:** <http://pages.cs.wisc.edu/~cs367-1>  
**Piazza:** <https://piazza.com/wisc/summer2014/cs367/>

**Instructor:** M. Hidayath Ansari ([ansari@cs.wisc.edu](mailto:ansari@cs.wisc.edu))  
**Office Hours:** Tuesday and Wednesday 12:30 – 1:30 PM + by appointment  
Medical Sciences Center 6795

**Teaching Assistant:** Aditya Singh ([asingh36@wisc.edu](mailto:asingh36@wisc.edu))  
**Office Hours:** Monday 1:30 – 2:30 PM (Computer Sciences 1301)  
**Lab Hours:** Thursday 3 – 4 PM (Mumble Lab, Computer Sciences)

**Course content (aka things you have to keep track of):**

- Lecture
- Online Readings
- Course website and Piazza
- Assignments (Written and Programming)
- Midterm and Final

Email [ansari@cs.wisc.edu](mailto:ansari@cs.wisc.edu) by Friday if you:

- have a conflict with the exam dates
- participate in religious observances that may interfere with course requirements
- have a VISA from the McBurney Center

## Course Work

### Homework Assignments (24%)

- 8 homeworks, 3% each
- no collaboration allowed
- not accepted late
- Submitted through Moodle

### Programming Assignments (26%)

- 4 programs, ~6-7% each
- no collaboration allowed
- Submitted through Moodle
- Late Policy:
  - up to 2 days late with penalty
  - 2 “free” late days cancel penalty

### Exams (45%)

- Midterm (20%): Thursday, July 11, in class
- Final (25%): Thursday, August 7, in class

## Agenda for Today

- Course Introduction and Logistics
- Sign up for Piazza!
- Abstraction, Interfaces, ADTs
- Sack ADT: Examples and Implementation
- Generics
- HW0 assigned

**What is this course about?**

# Abstraction, Interfaces, and ADTs

## Example: Sack ADT

### Conceptual picture

### Public interface/operations

```
void add(Object item)
Object remove() throws NoSuchElementException
boolean isEmpty()
```

### Implementation challenges

## Using the Sack ADT: Example 1

Write a code fragment to put the numbers 0 through 99 into the sack.

```
Sack s = new Sack();
```

## Autoboxing

## Using the Sack ADT: Example 2

Complete the `printSack()` method: prints the contents of the Sack parameter.

*Challenge:* Implement the method so it doesn't change the sack's contents.

```
public void printSack(Sack s) {
```

## **Implementing Sack ADT**

**Storage?**

**What kind of elements?**

**Basic structure of Sack ADT implementation**

### Using the Sack ADT: Example 3

Why doesn't the following code compile? Suppose `sack` is an instance of Sack ADT above and is not empty.

```
for (int i = 0; i < 10; i++) sack.add(new Die());

while (!sack.isEmpty()) {
    Die myDie = sack.remove();
    myDie.roll();
}
```

**What makes software good?**