

LEC 17

CSE 122

Advanced JUnit

BEFORE WE START

*Slido vote & chat with neighbors:**What is your favorite shape of pasta?*Music: [122 26sp Lecture Jams](#) **Instructor:** Elba S Garza

<b>TAs:</b>	David	Caleb	Cole	Yang
	William	Neha	Blake R.	Cady
	Dani	Wesley	Carson	Diya
	Rohan	Isis	Sushma	Elsa Garza
	Andrew	Colin	Connor	
	Ava	Naomi	Mahima	
	Shreyank	Hanna	Nicolae	
	Nicole	Blake P.	Ivory	

Questions during Class?

Raise hand or send here

sli.do #cse122



# Lecture Outline

- **Announcements** ◀
- JUnit Recap
- Card & BattleManager

# Announcements

- Creative Project 3 (C3) will be released later today
  - Focused on JUnit/Testing
- Final Exam: **Tuesday, June 9th 2:30-4:20 PM**
  - KNE 120 (here)
- More information on the Final Exam will be posted on the course website later tonight!
- JUnit “virtual section” for JUnit content, linked on Ed

# Lecture Outline

- Announcements
- **JUnit Recap** ◀
- Card & BattleManager

# JUnit Recap (1/2)

- JUnit – a unit testing framework for the Java language
- Unit Test – testing a specific part of your program
- Method Annotations ('@' Symbols)
  - `@Test`
    - Tells Java to treat the method like a JUnit test
  - `@DisplayName`
    - Lets us name our test cases
- Assertion Methods
  - `assertEquals()`
  - `assertTrue()`
  - `assertFalse()`
  - ...

# JUnit Recap (2/2)

- **Specification Testing** – You do not need to look at the code to test
  - Tests are based on what the end goals of the methods are
  - Can be written in advance before the code
  - We will use specification testing more (like in C3)
- **Clearbox Testing** – Specifically using your implementation
  - Write after you have written your code



# Practice : Think



sli.do #cse122

**Which of the following correctly compares two lists have the same elements in the same order?**

- A. `assertSame(list1, list2);`
- B. `assertArrayEquals(list1, list2);`
- C. `assertTrue(list1.equals(list2));`
- D. `assertTrue(list1 == list2);`
- E. `assertFalse(list1 == list2);`



# Practice : Pair



sli.do #cse122

**Which of the following correctly compares two lists have the same elements in the same order?**

- A. `assertSame(list1, list2);`
- B. `assertArrayEquals(list1, list2);`
- C. `assertTrue(list1.equals(list2));`
- D. `assertTrue(list1 == list2);`
- E. `assertFalse(list1 == list2);`

# Lecture Outline

- Announcements
- JUnit Recap
- **Card & BattleManager** 

# Card Class

- `public Card(int value, String suit)`
  - Throws an `IllegalArgumentException` if `value` or `suit` is invalid
- `public String getSuit()`
- `public int getValue()`
- `public String toString()`
- `public boolean equals(Object other)`



# BattleManager Class

- `public BattleManager()`
- `public BattleManager(List<Card> deck1,  
List<Card> deck2)`
- `public int getPlayer1DeckSize()`
- `public int getPlayer2DeckSize()`
- `public void deal()`
- `public boolean gameOver()`
- `public void play()`
  - Throws an `IllegalStateException` if either hand is empty

