Essence and the Basics of Software Engineering

Idaho State University | Computer Science

Dr. Isaac Griffith

CS 2263
Department of Computer Science
Idaho State University

ROAR

Daily Quiz

- Starting at 14:02
    - You will have 5 minutes
    - You may not use your notes, books, lectures, or the internet
    - You may not help each other

Passcode

1671

ROAR

Individual Practicum

Part 1

Write a Java class called `LongInteger` as per the following specifications

- Objects of the class store integers that can be as long as 50 digits.

- The class must have the following constructors and methods

  - `public LongInteger()` - Sets the integer to 0

  - `public LongInteger(int[] otherDigits)` - Sets the integer to the given integer represented by the parameter. A copy of otherDigits must be made to prevent accidental changes.

  - `public LongInteger(int number)` - Sets the integer to the value given in the parameter

  - `public void readIn()` - reads in the integer from the keyboard. You can assume that only digits will be entered

  - `public LongInteger add(int number)` - Adds number to the integer represented by this object and returns the result

  - `public LongInteger add(LongInteger number)` - Adds number to the integer represented by this object and returns the result

  - `public String toString()` - returns a String representation of the integer.

- Use an array of 50 ints to store the digits of the number

Part 2

- Study the interface `Extendable` given below:

```java
public interface Extendable {
    boolean append(char c);
    boolean append(char[] sequence);
}
```

- `append(char c)` - appends a character to the object (or, more precisely the object's class) that implements this interface.

- `append(char[] sequence)` - appends all characters in the array to this object.

  - If there is no space in the object to append, the method returns *false*; otherwise they return true.

- Write code for the class `SimpleBuffer` that implements the above interface

  - Has a constructor with the following signature.
    ```java
    public SimpleBuffer(int size)
    ```
  - The class has two fields

    - one which stores the char array

    - the other stores the number of elements actually filled in the array

  - Must also implement the `toString` method which returns the correct String representation of the char array.

  - It should also implement the `equals` method such that two buffers are equal if and only if they contain the same set of characters

ROAR