



Object-Oriented Programming

CS 1331

Recitation 13

Announcements

- Final exam is coming soon!
 - Section A (10:10am): Friday, December 6th @ 8:00am
 - Section B (9:05am): Wednesday, December 11th @ 8:00am
 - Section C (12:20pm): Friday, December 6th @ 11:20am
- Office Hours are closed
- Piazza will be disabled after the first final exam
- CLOS Bonuses: .5% bump if 90% complete
- HW13-16 regrades are due on Gradescope by 12/7 NOON

Collections

- Java has built in ADTs and data structures for us to use
- Java Collections Framework: a unified architecture for representing and manipulating collections, allowing groups of data to be utilized independently of implementation details.
- Base of the framework: Collection<E> interface
 - Required methods: `contains(Object o)`, `containsAll(Collection<?> c)`, `equals(Object o)`, `hashCode()`, `isEmpty()`, `iterator()`, `size()`, `toArray()`, `toArray(T[] a)`

Data Structures

- Set<E>:
 - Implements Collection
 - Instantiable extension: HashSet<E>
- List<E>:
 - Implements Collection
 - Instantiable extensions: ArrayList<E>, LinkedList<E>
- Map<K, V>
 - Does NOT implement Collection
 - Instantiable extension: HashMap<K, V>

HashSet

- Holds distinct elements in no particular order
- Cannot access elements by index
- If attempt is made to add a duplicate element to the set as determined by the equals method, it will not add it
- Very efficient for checking if some element is a member of a particular group

ArrayList & LinkedList

- ArrayList
 - Holds ordered elements adjacently in memory
 - Backed by an array
 - Each element has an index and accessing by index is very efficient
 - Adding to the back is more efficient than adding to the front
- LinkedList
 - Holds ordered elements but may be scattered in memory
 - Each element has a variable “next” that gives the location of the next element in the list
 - No indices, have to iterate through the list in order to access an element with an arbitrary position in the list
 - Adding to the front is always efficient because of the head pointer, but if the list has a tail pointer then adding to the back is also efficient

Iterator

- An object that we use with a collection with the purpose of accessing each of its elements
- Iterable<E> interface
 - iterator(): returns an object that implements Iterator<E> interface
- Iterator<E> interface
 - next(): returns the next element
 - hasNext(): returns whether or not there is a next element
 - remove(): default method that removes from the collection the last element that was returned by next()

HashMap

- A concrete implementation of Java's Map interface
- Used to store Key, Value pairs
 - For example: `HashMap<String, Integer>` can be used to store a person's phone number (value) by their name (key)
- Allows duplicate values but not duplicate keys
 - Two keys cannot be equal, but two values can be equal
- Items have no particular ordering, similar to a set
- Large benefit of HashMap is its amortized $O(1)$ accessing due to using hashing to store its key, value pairs
- Notable methods:
 - `V` `get(Object key)`
 - `V` `put(K key, V value)`
 - `V` `remove(Object key)` -Removes the mapping for the specified key from this map if present.

Hashing

- Used to quickly store and retrieve values
- Why use hashing when we could use an array?
 - Allows us to compress storage
 - Say we wanted to store people (value) by their SSN (key)
 - We would need 10^9 spots in an array to store every possible SSN. We could use 100 spots instead and hash the SSN using $\% 100$
- Function that converts a key into an index is called a hashing function
 - Usually uses division or modulus
 - Finds the hashcode of a key and then compresses the hashcode into the index

Resources to Help Study

- All coding from recitation and lecture are posted on Canvas (look under Files and Modules)
- All practice exams/worksheets from previous exams are on Canvas
- Exam Resources folder on Canvas has past exams from semesters before
- Living Schedule has all topics covered that will be on the Final
- Professors post under Announcements on Canvas helpful book problems

Final Exam Review Kahoot

- <https://play.kahoot.it/v2/?quizId=6e1534d0-411b-4416-9c87-33d79305d186>