Iterators and binary search

2/17/16
Announcements

• Reading:
  – Friday: Java Interlude 8 and Chapter 19 (Dictionaries)
  – Monday: Chapter 21 (Hashing)

• Homework out soon!

• Friday’s quiz: solve some design problems
Iterator ADT

• boolean hasNext()
• T next()       //returns and advances
• void remove()  //optional: removes most recent

• ListIterator: hasNext, previous, nextIndex, previousIndex, add(T), set(T) (optional)
Inheritance

• extends: makes class a subtype of another
  – terminology: subclass and superclass
  – subclass has all attributes and methods of superclass
    • subclass can override methods it wants to change
  – references to subclass can be stored in superclass-typed variables

As a general rule, be as restrictive as possible
Inheritance

- **extends**: makes class a subtype of another
  - terminology: subclass and superclass
  - subclass has all attributes and methods of superclass
    - subclass can override methods it wants to change
  - references to subclass can be stored in superclass-typed variables

- **access control**
  - public: anyone can access it
  - private: can only access from within the class
  - protected: accessible from within class or a subclass

As a general rule, be as restrictive as possible
Which of the following lines has a syntax error?

class Person {
    private String name;
}
class Student extends Person {
    private double credits;
    private double qualityPts;

    public double gpa() {
        return qualityPts / credits;
    }
    public String toString() {
        return name + " (GPA: \n + gpa() + \n)"; //B
    }
    ...
}

Person p = new Student(); //C
System.out.println(p.gpa()); //D

//E: Not exactly one of the above
Which of the following lines has a syntax error?

class Person {
    private String name;
}

class Student extends Person {
    private double credits;
    private double qualityPts;

    public double gpa() {
        return qualityPts / credits;
    }

    public String toString() {
        return name + " (GPA: " + gpa() + ")";
    }

    ...  
}

Person p = new Student();  //C
System.out.println(p.gpa());  //D

//E: Not exactly one of the above (B & D)
New and improved Set ADT

• Represents unordered set of values
  \{“hello”, “there”, “=)”, “CS 142”\}

• **No duplicate elements**

• Supports:
  – boolean add(value)    //returns whether it was new
  – boolean contains(value)
  – Iterator iterator()

Maybe also: size, remove, clear