

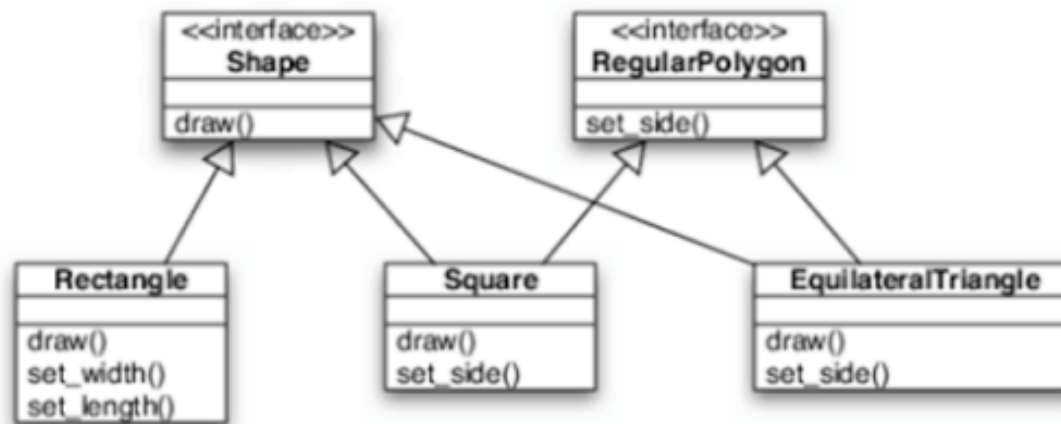
Agenda: Discussion Week 10

JUNE 1, 2009

Reminder: Quiz 4 on Tuesday, June 2

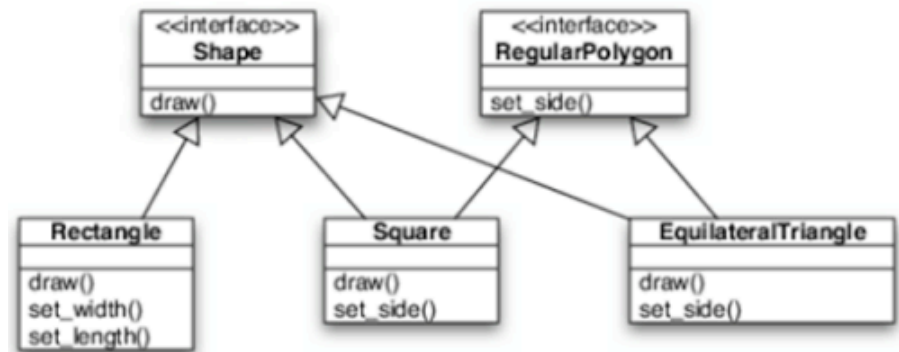
- Inheritance
- Algorithm runtime analysis
- PSA 5: Critters
- Suggestions?

Inheritance example



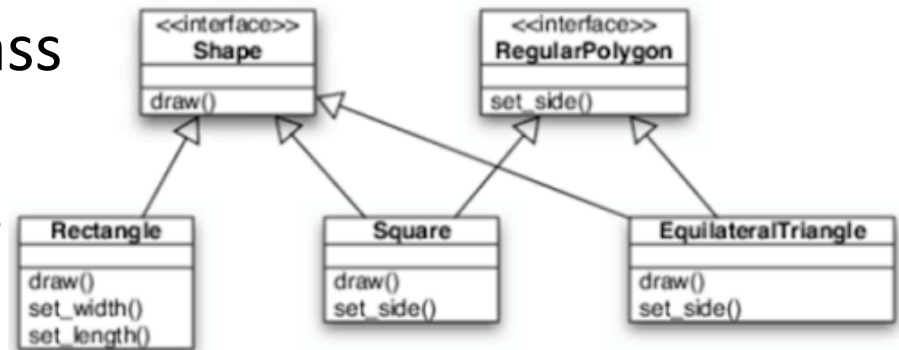
Inheritance (1)

- Inheritance – single or multiple?
 - Can extend ≤ 1 class
 - Can implement multiple interfaces
- Superclass (base)
- Subclass (derived)
- Access modifiers
 - private - class
 - default (no modifier) – class, package
 - protected – class, package, subclass
 - public – class, package, subclass, world



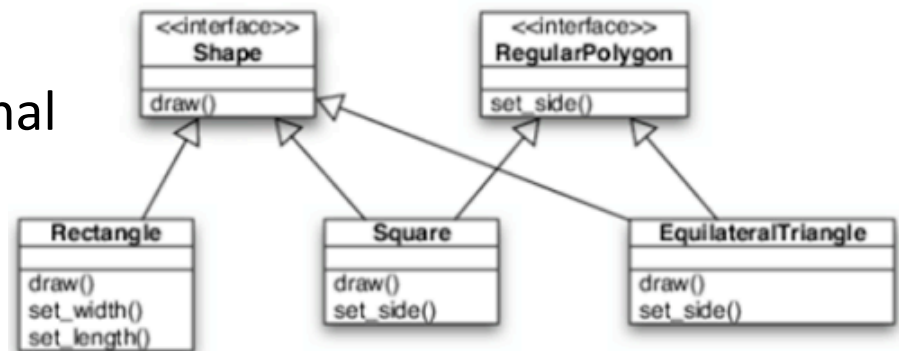
Inheritance (2)

- super / this
 - Constructors: super(), this()
 - Fields: super.field, this.field
 - “this” object
- Fields / methods in superclass
 - Can subclass access?
- Fields / methods in subclass
 - Can superclass access?
- Parameter passing
 - Superclass / subclass parameters
 - “is-a” relationship



Inheritance (3)

- Field hiding
 - Defining field with same name as superclass' field
 - Not recommended – makes code difficult to read
- Method overriding (instance) and hiding (static)
 - Defining method with same signature as superclass' method
 - Final
 - Can't override
 - Static methods implicitly final
 - When?
 - Access?



Examples: Overriding methods

- toString

```
public String toString() {  
    return "A string that represents this object";  
}
```

- equals

```
public boolean equals(Object o) {  
    // Return true if this equals o  
    // Otherwise return false  
}
```

Example: toString

```
public class Rectangle {  
    int width;  
    int height;  
  
    public String toString() {  
        // Print the width and height  
        // Example: "width is 5, height is 7";  
    }  
}
```

Algorithm runtime analysis

- Worst case
 - Which input value(s) will produce the worst case (i.e. longest) running time for the algorithm?
- What is the worst case?
 - The worst case depends on the algorithm
 - How to specify?
 - Input value(s)
 - Running time: iterations (loop), method calls (recursion)
 - How to evaluate?
 - Trace program execution on input values
- Examples: Search
 - Linear search, recursive linear search, binary search

Tracing program execution

- Examples
 - Trace values on each iteration of loop
 - Trace values on each call to recursive method

Example: Recursive linear search (starting at end of array)

```
public int linearSearch(int[] intArray, int toFind, int
    index) {
    if (intArray[index] == toFind)
        return index;
    if (index == 0)
        return -1;
    else return linearSearch(intArray, toFind, index-1);
}
```

Example: Binary search

```
public static int binarySearch(int[] intArray, int start, int end, String toFind)
{
    if (start > end)
        return -1;
    if (start == end) {
        if (intArray[start] == toFind)
            return start;
        else
            return -1;
    }
    int mid = (start+end)/2;
    if (toFind == intArray[mid])
        return mid;
    if (toFind > intArray[mid])
        return binarySearch(intArray, start, mid-1, toFind);
    else
        return binarySearch(intArray, mid+1, end, toFind);
}
```

PSA 5

- PSA 5: Critters (inheritance)
 - Due Tuesday, June 2, 11:59pm
 - <http://cs.ucsd.edu/classes/sp09/cse8b/psas/PSA5/PSA5.html>
 - Tournament in class Thursday, June 4
- Implement as specified
 - Bear
 - Lion
 - Tiger
- Implement as you choose (for tournament)
 - Triton
- Questions?