## **Section Handout**

## **OO runtime organization**

```
class A {
           int x;
          static int y;
          public foo() { ... }
public bar(int x, int y) { ... }
public static baz() { ... }
}
class B extends A {
          int x;
          int z;
          public foo() { ... }
}
static void main() {
          A a1, a2;
          B b1;
          a1 = new A();
          a2 = new B();
          b1 = new B();
}
```

- Show the layout of the structures pointed to by a1 and a2. (Show where all of the instance and static variables are stored, the contents of the dispatch tables, which structures are shared, and which are not)
- What assembly code is needed for a method call a1.bar(1,2)? (Don't have to be precise, just state what needs to be done in order to call the method)
- In Java, methods which are private are not accessible by subclasses. (For example, if bar() in class A was private, then B could not call bar() in any of its methods.) However we still keep bar() in the dispatch table. Why is this not wasted space?
- •

Last modified: Tue Nov 11 19:06:34 Pacific Standard Time 2003