

CS 100 Exercise 13: Inheritance and Shadowing
Summer 2001
Due in lecture, Tuesday July 31

Name: _____ CUid: _____

Read the following code. Then, on the back of this sheet, **draw a class hierarchy** that includes classes **A** and **B** (the boxes should include variables and methods!). Then, **show the output** that would be produced if you set the Target to Ex13. This is a short exercise, but requires a lot of thought. If you don't have a headache when you're done, you haven't thought deeply enough about what's going on.

```
class A {
    public int x;
    public A(int x) { this.x = x; }
    public int getX() { System.out.print("class A: "); return x; }
}
class B extends A {
    public int x;
    public B(int x) { super(x*10); this.x = x; }
    public int getX() { System.out.print("class B: "); return x; }
    public int getSuperX()
    { System.out.print("class B super: "); return super.getX(); }
}

public class Ex13 {
    public static void main(String[] args) {
        A a1 = new A(5);
        System.out.print("(a1) ");          System.out.println(a1.getX());
        B b = new B(3);
        System.out.print("(b) ");           System.out.println(b.getX());
        System.out.print("(b) ");           System.out.println(b.getSuperX());
        A a2 = new B(12);
        System.out.print("(a2) ");           System.out.println(a2.getX());
        // Can't make the following call, because it doesn't exist in A
        // System.out.print("(a2) ");         System.out.println(a2.getSuperX());
        A a3 = b;
        // Now a3 and b point to the same thing.
        System.out.print("(b) ");           System.out.println(b.getX());
        System.out.print("(b) ");           System.out.println(b.getSuperX());
        b.x = 1000;
        System.out.print("(b) ");           System.out.println(b.getX());
        System.out.print("(b) ");           System.out.println(b.getSuperX());
        a3.x = 2000;
        System.out.print("(b) ");           System.out.println(b.getX());
        System.out.print("(b) ");           System.out.println(b.getSuperX());
    }
}
```