Lec 8

Memory, Eclipse, JUnit Testing
Draw the order of methods on the stack up to the indicated point of program execution, method6. Note: the stack may not fill completely, also Stacks fill from the bottom to the top:

public class Driver {
    public static void main(String [] args) {
        SomeClass sc = new SomeClass();
        method1();
        method4();
    }
}

public class SomeClass {
    public void method1() {
        method5();
    }
    public void method2() {
        return;
    }
    public void method3() {
        method6();
    }
    public void method4() {
        method2();
    }
    public void method5() {
        method3();
    }
    public void method6() {
        /* What is the stack here? */
    }
}
Draw the memory models for the variables of main()

```java
public class Ex {
    public static void main(…) {
        int i = 0;
        int[] array = new int[10];
        Chalk c1 = new Chalk(“white”);
        Chalk c2 = new Chalk(“green”);
        c3 = c1;
    }
}
```

```java
public class Chalk {
    private String color;
    public Chalk(String c) {
        color = c;
    }
}
```
Draw the memory models for Driver & Animal. What is the output?

```java
public class Driver {
    public static void main(…) {
        Animal a1 = new Animal("blue");
        Animal a2 = new Animal("purple");
        System.out.println(a1.getColor());
        System.out.println(a2.getColor());
    }
}
```

```java
public class Animal {
    private static String color;

    public Animal (String c) {
        color = c;
    }

    public String getColor() {
        return color;
    }
}
```

a) blue, blue  
b) blue, purple  
c) purple, blue  
d) purple, purple
IDE
Phone

• Write a program to store phone contacts
  – Should be able to add contacts, remove contacts, and find a contact

• Ex. Usage
  1) Add contact
  2) Remove contact
  3) Find contact
  4) Quit

  Section: 1
  Contacts name: Adam Jundt
  Contacts number: 123
  Adam Jundt successfully added!
  1) Add contact
  2) Remove contact
  3) Find contact
  4) Quit

  Section: 3
  Contacts name: Adam Jundt
  Adam Jundt: 123
  1) Add contact
  2) Remove contact
  3) Find contact
  4) Quit

  Section: 2
  Contacts name: Adam Jundt
  Adam Jundt successfully removed
JUnit Testing

• `assertTrue(condition)`
• `assertEquals(expected, actual)`
• `assertArrayEquals(expected, actual)`
  – Works for 2d arrays too (HW4)
Eclipse

- Ctrl-space - auto suggest
- Debugger
- JUnit Testing