Lab 7 Written Portion

All of the questions in this portion make use of the following class definition:

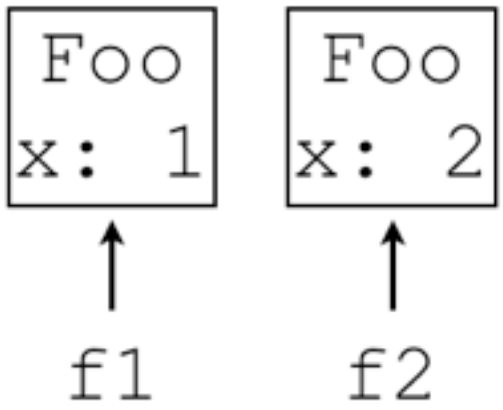
```
public class Foo {
  private int x; // instance variable

  // constructor
  public Foo(int y) {
    x = y; // set instance variable
  }

  // setter
  public void setX(int newX) {
    x = newX;
  }

  public String toString() {
    return "" + x;
  }
}
```

Memory Representation

The following questions require you to write out how a snippet of code will "look" in memory after being executed. Some of them have already been done for you.

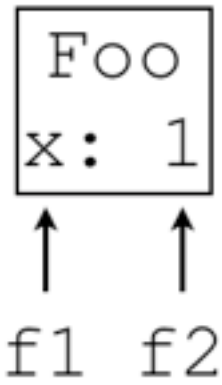
1.)

```
Foo f1 = new Foo(1);
Foo f2 = new Foo(2);
```



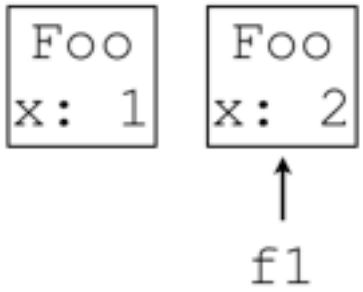
2.)

```
Foo f1 = new Foo(1);
Foo f2 = f1;
```



3.)

```
Foo f1 = new Foo(1);
f1 = new Foo(2);
```

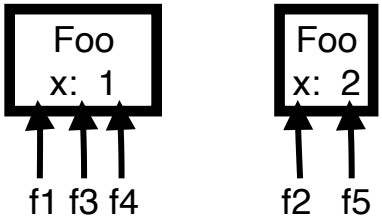


4.)

```
Foo f1 = new Foo(1);
Foo f2 = new Foo(2);
Foo f3 = new Foo(3);
```

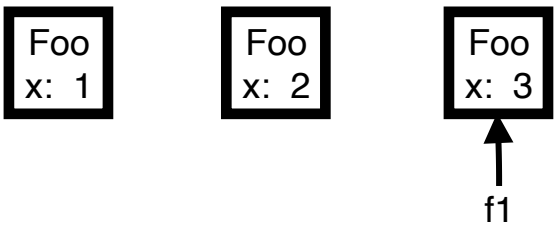
5.)

```
Foo f1 = new Foo(1);
Foo f2 = new Foo(2);
Foo f3 = f1;
Foo f4 = f3;
Foo f5 = f2;
```



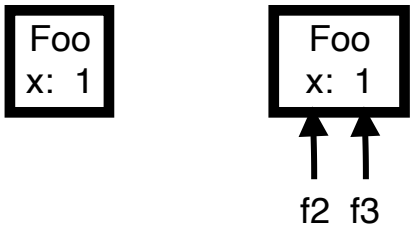
6.)

```
Foo f1 = new Foo(1);
Foo f2 = new Foo(2);
f1 = new Foo(3);
```



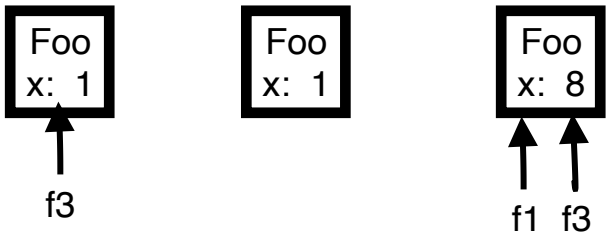
7.)

```
// Hint: new always
// creates a new object
Foo f1 = new Foo(1);
Foo f2 = new Foo(1);
Foo f3 = f2;
```



8.)

```
Foo f1 = new Foo(1);
Foo f2 = new Foo(2);
Foo f3 = f1;
f3.setX(8);
```



Code Output

The following questions ask you what the output of a snippet of code is, starting execution from `main`. The first one has been done for you as an example.

9.)

```
public static void method1(int x) {
  // empty method
}

public static void main(String[] args) {
  int x = 7;
  System.out.println(x);
  method1(x);
  System.out.println(x);
}
```

OUTPUT:

```
7
7
```

10.)

```
public static void method2(int x) {
  x = 14;
}

public static void main(String[] args) {
  int x = 7;
  System.out.println(x);
  method2(x);
  System.out.println(x);
}
```

OUTPUT:

```
  7
  7
```

11.)

```
public static void method3(Foo x) {
  // empty method
}

public static void main(String[] args) {
  Foo x = new Foo(7);
  System.out.println(x);
  method3(x);
  System.out.println(x);
}
```

OUTPUT:

```
7
7
```

12.)

```
public static void method4(Foo x) {
  x = new Foo(8);
}

public static void main(String[] args) {
  Foo x = new Foo(7);
  System.out.println(x);
  method4(x);
  System.out.println(x);
}
```

OUTPUT:

```
7
7
```

13.)

```
public static void method5(Foo x) {
  x.setX(8)
}

public static void main(String[] args) {
  Foo x = new Foo(7);
  System.out.println(x);
  method5(x);
  System.out.println(x);
}
```

OUTPUT:

7
8

14.)

```
public static void method6(Foo x) {
  x = new Foo(8);
  x.setX(9)
}

public static void main(String[] args) {
  Foo x = new Foo(7);
  System.out.println(x);
  method6(x);
  System.out.println(x);
}
```

OUTPUT:

7
7

15.)

```
public static void method7(Foo x) {
  x.setX(9)
  x = new Foo(8);
}

public static void main(String[] args) {
  Foo x = new Foo(7);
  System.out.println(x);
  method7(x);
  System.out.println(x);
}
```

OUTPUT:

7
9

16.)

```
public static void method8(Foo x) {
  x.setX(9)
  x = new Foo(8);
  x.setX(10);
}

public static void main(String[] args) {
  Foo x = new Foo(7);
  System.out.println(x);
  method8(x);
  System.out.println(x);
}
```

OUTPUT:

7
9