COP 3330: Object-Oriented Programming
Summer 2011

EXAM #1 – In Class Practice

Instructor : Dr. Mark Llewellyn
markl@cs.ucf.edu
HEC 236, 407-823-2790
http://www.cs.ucf.edu/courses/cop3330/sum2011

Department of Electrical Engineering and Computer Science
Computer Science Division
University of Central Florida

Static vs. Class Variables and Methods – In-Class Practice Problem

First – identify which statements are legal and which are illegal.

Second – fix the illegal statements.

Third – trace the program execution and show the output.

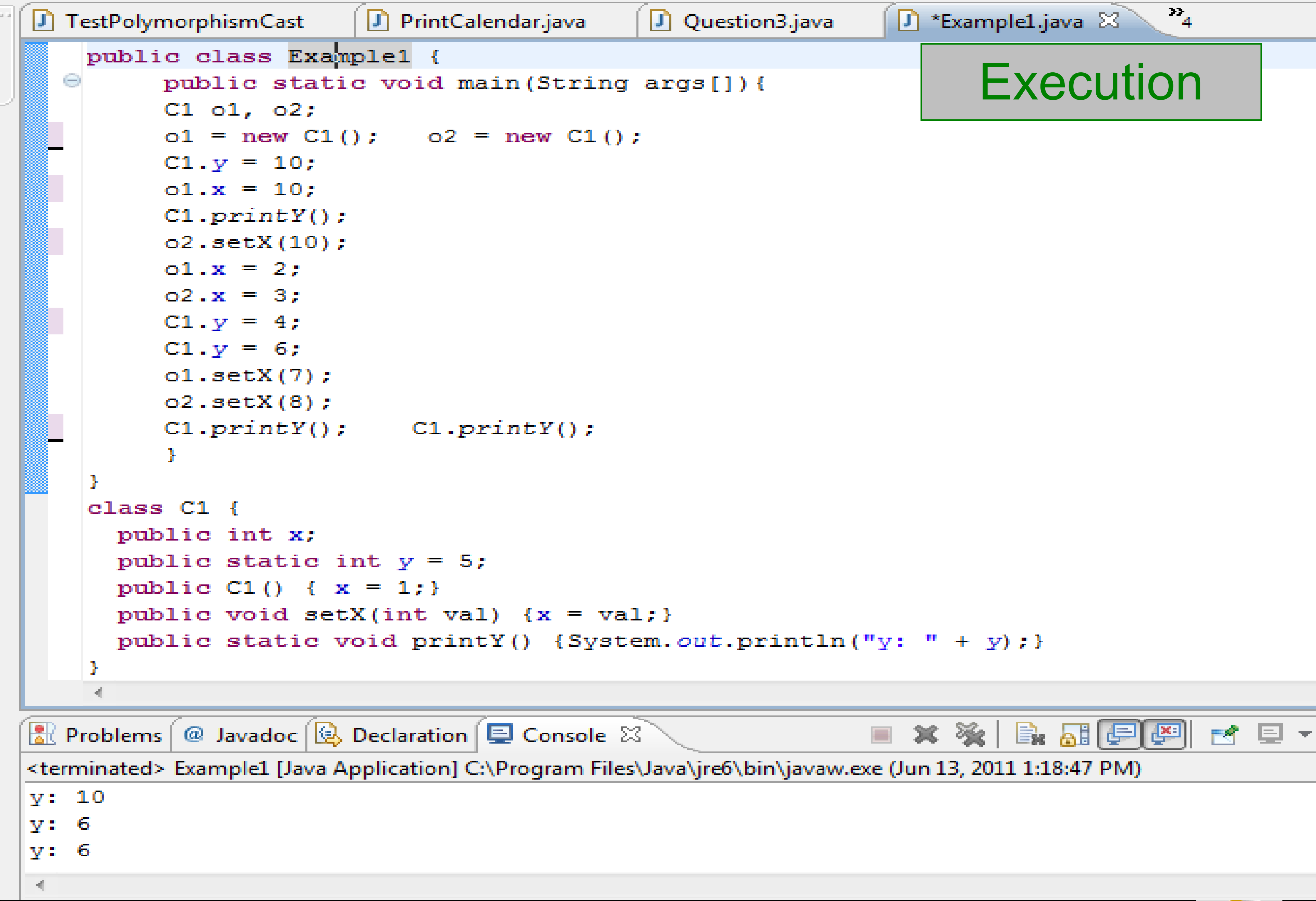
```
public class Example1 {
     public static void main(String args[]){
     C1 o1, o2;
     C1.y = 10;
     C1.x = 10;
     C1.printY();
     C1.setX(10);
     o1 = new C1();
     o2 = new C1();
     o1.x = 2;
     o2.x = 3;
     o1.y = 4;
     C1.y = 6;
     o1.setX(7);
     o2.setX(8);
     C1.printY();
     o2.printY();
     }
}
class C1 {
  public int x;
  public static int y = 5;
  public C1() { x = 1;}
  public void setX(int val) {x = val;}
  public static void printY() {System.out.println("y: " + y);}
}
```

```java
public class Example1 {
     public static void main(String args[]){
     C1 o1, o2;
     C1.y = 10;
     C1.x = 10; //cannot make a static reference to a non-static field
     C1.printY();
     //C1.setX(10); //cannot make a static reference to a non-static method
     o1 = new C1();
     o2 = new C1();
     o1.x = 2;
     o2.x = 3;
     o1.y = 4; //static variable - should be accessed thru the class
     C1.y = 6;
     o1.setX(7);
     o2.setX(8);
     C1.printY();
     o2.printY(); // static method should be accessed thru the class
     }
}
class C1 {
  public int x;
  public static int y = 5;
  public C1() { x = 1;}
  public void setX(int val) {x = val;}
  public static void printY() {System.out.println("y: " + y);}
}
```

The Fix

```
public class Example1 {
     public static void main(String args[]){
     C1 o1, o2;
    o1 = new C1();
    o2 = new C1();
    C1.y = 10;
     o1.x = 10; //make reference to instance variable thru an instance
     C1.printY();
     o2.setX(10); //invoke instance method thru an instance
     o1.x = 2;
     o2.x = 3;
     C1.y = 4; //static variable referenced/accessed thru the class
     C1.y = 6;
     o1.setX(7);
     o2.setX(8);
     C1.printY();
     C1.printY(); // static method invoked thru the class
     }
}
class C1 {
  public int x;
  public static int y = 5;
  public C1() { x = 1;}
  public void setX(int val) {x = val;}
  public static void printY() {System.out.println("y: " + y);}
}
```

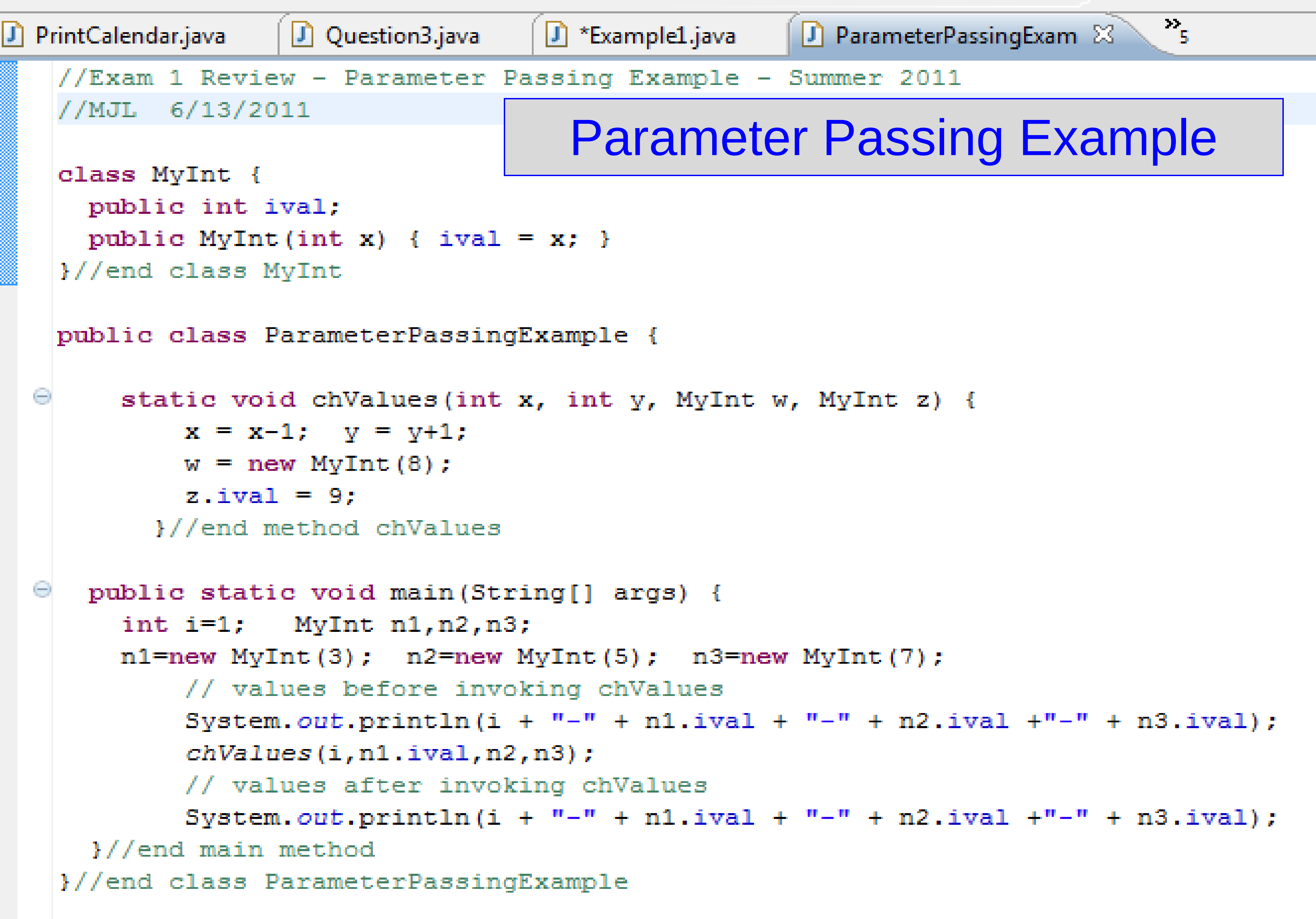
Execution



```
public class Example1 {
        public static void main(String args[]){
        C1 o1, o2;
        o1 = new C1();    o2 = new C1();
        C1.y = 10;
        o1.x = 10;
        C1.printY();
        o2.setX(10);
        o1.x = 2;
        o2.x = 3;
        C1.y = 4;
        C1.y = 6;
        o1.setX(7);
        o2.setX(8);
        C1.printY();     C1.printY();
        }
}
class C1 {
   public int x;
   public static int y = 5;
   public C1() { x = 1;}
   public void setX(int val) {x = val;}
   public static void printY() {System.out.println("y: " + y);}
}
```

```
<terminated> Example1 [Java Application] C:\Program Files\Java\jre6\bin\javaw.exe (Jun 13, 2011 1:18:47 PM)
y: 10
y: 6
y: 6
```

Parameter Passing Example

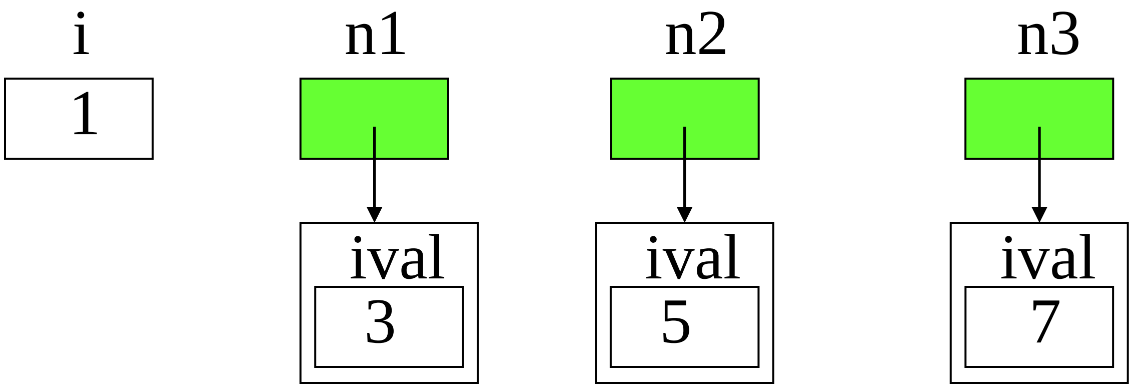


```java
//Exam 1 Review - Parameter Passing Example - Summer 2011
//MJL  6/13/2011

class MyInt {
  public int ival;
  public MyInt(int x) { ival = x; }
}//end class MyInt

public class ParameterPassingExample {

    static void chValues(int x, int y, MyInt w, MyInt z) {
        x = x-1;  y = y+1;
        w = new MyInt(8);
        z.ival = 9;
      }//end method chValues

  public static void main(String[] args) {
    int i=1;   MyInt n1,n2,n3;
    n1=new MyInt(3);  n2=new MyInt(5);  n3=new MyInt(7);
        // values before invoking chValues
        System.out.println(i + "-" + n1.ival + "-" + n2.ival +"-" + n3.ival);
        chValues(i,n1.ival,n2,n3);
        // values after invoking chValues
        System.out.println(i + "-" + n1.ival + "-" + n2.ival +"-" + n3.ival);
  }//end main method
}//end class ParameterPassingExample
```

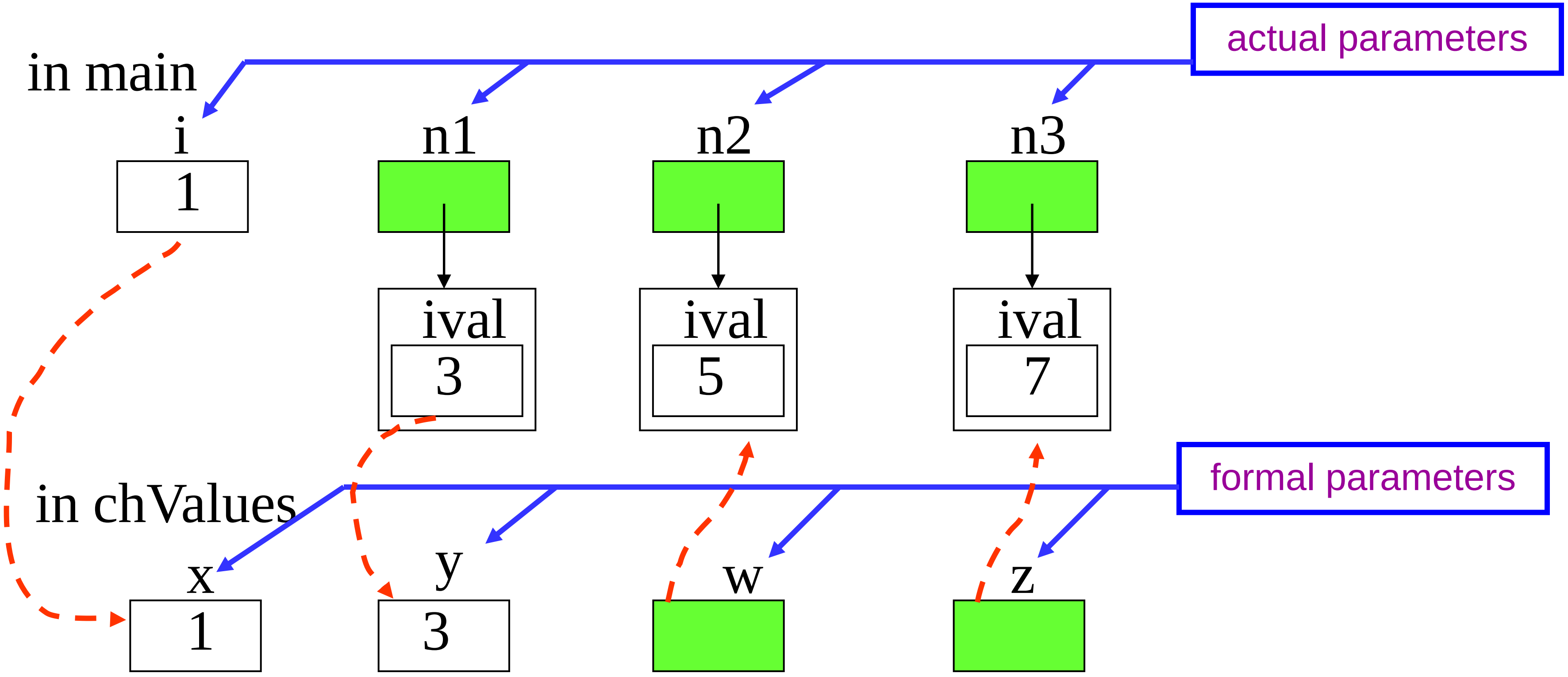
Parameter Passing Example

Step 1 - Execution begins in Main

in main before invocation of chValues



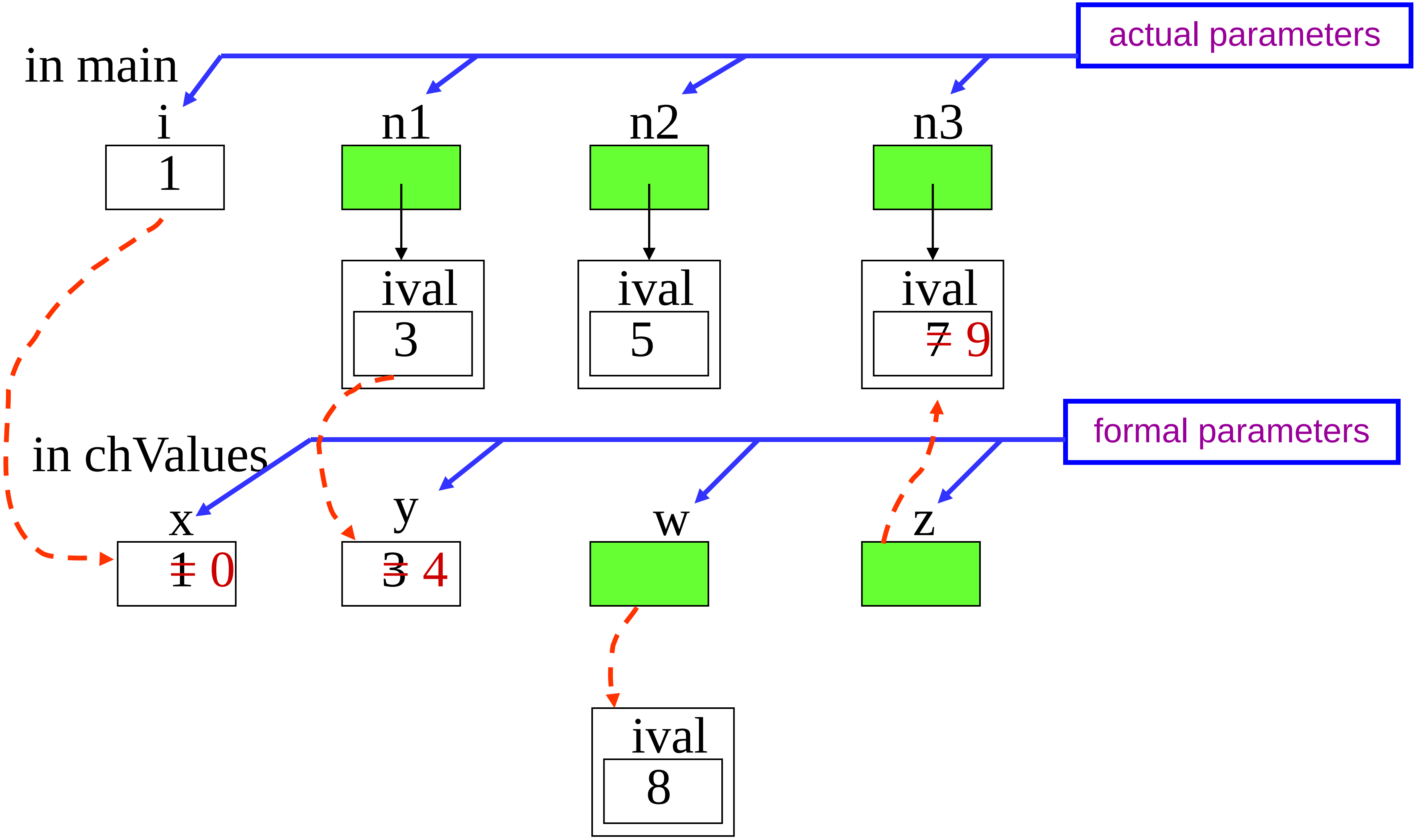
Step 2 – Invoke Method chvalues (i,n1.ival,n2,n3)



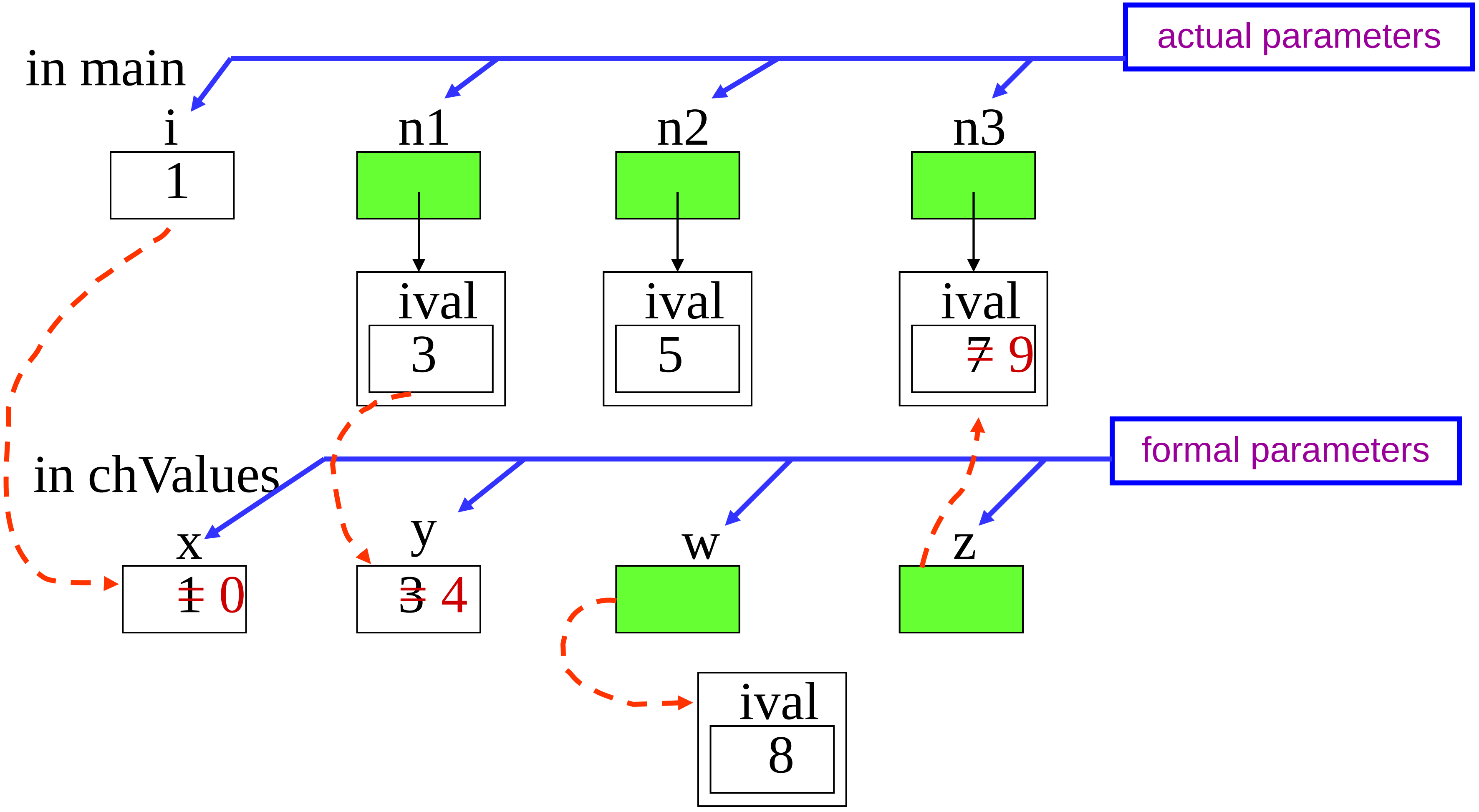
Red dotted lines indicate the passing of parameter values.

Notice that formal parameters x and y are primitive types and thus copies of the actual parameters are passed (pass by value) while w and z are objects and thus references to the objects are passed (pass by reference).

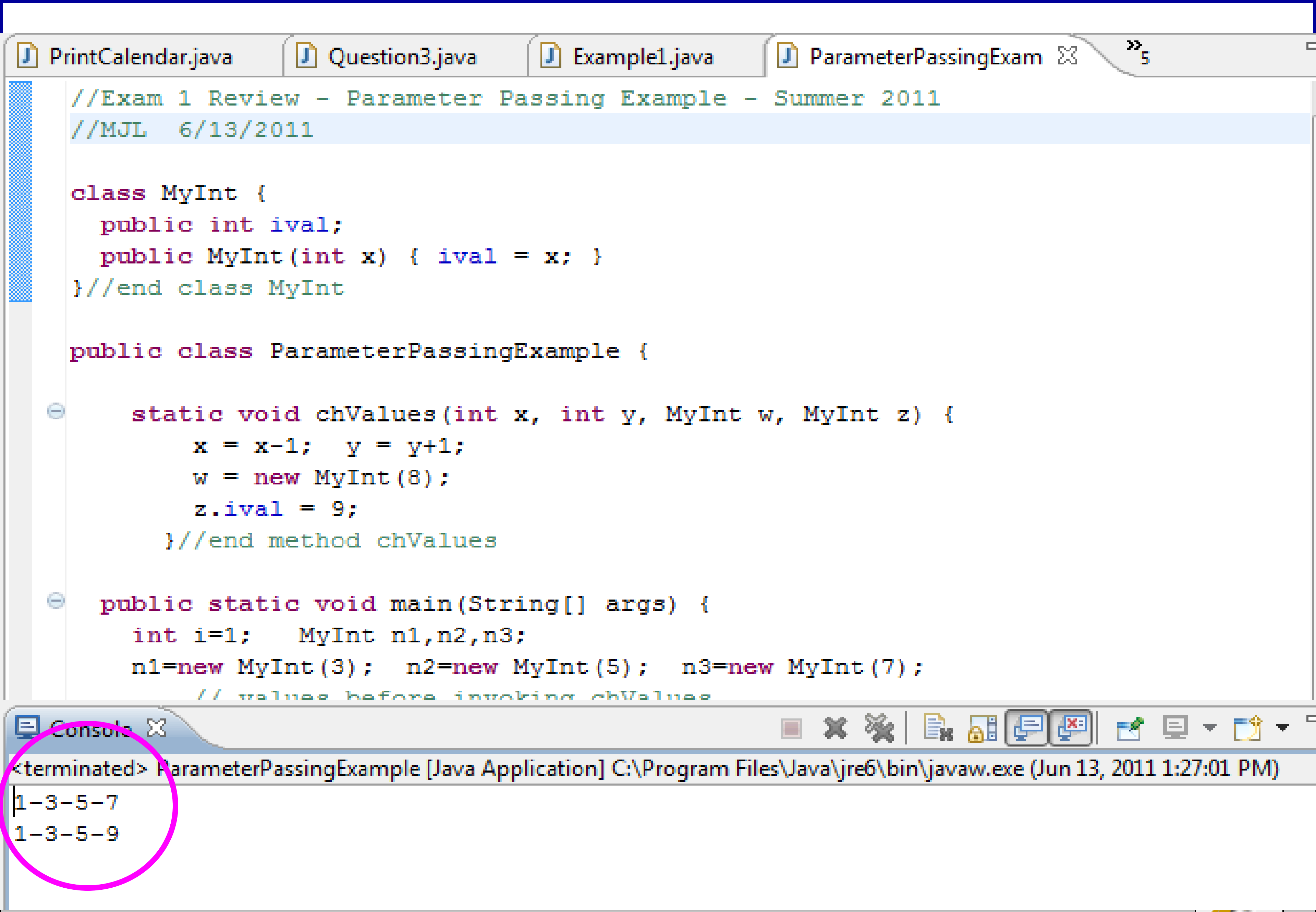
Step 3 – Method chValues Executes



Step 4 – Return to Main Method – Print Results



Output: 1 – 3 – 5 – 9



```
//Exam 1 Review - Parameter Passing Example - Summer 2011
//MJL  6/13/2011

class MyInt {
  public int ival;
  public MyInt(int x) { ival = x; }
}//end class MyInt

public class ParameterPassingExample {

    static void chValues(int x, int y, MyInt w, MyInt z) {
        x = x-1;  y = y+1;
        w = new MyInt(8);
        z.ival = 9;
      }//end method chValues

  public static void main(String[] args) {
    int i=1;   MyInt n1,n2,n3;
    n1=new MyInt(3);  n2=new MyInt(5);  n3=new MyInt(7);
        // values before invoking chValues
```

```
<terminated> ParameterPassingExample [Java Application] C:\Program Files\Java\jre6\bin\javaw.exe (Jun 13, 2011 1:27:01 PM)
1-3-5-7
1-3-5-9
```