

TOPIC 3 EXERCISES

Tracing Exercises

1. Trace each of these Java programs step by step. Show exactly what is printed by each program.

```
(a) public class prob3_1a {
    public static void main(String[] args)
    {
        int p,q=2;

        System.out.println("q p ");
        while (q <= 6) {
            p = 2 * q + 3;
            System.out.println(q + " " + p);
            q +=2;
        }
    }
}
```

```
(b) public class prob3_1b {
    public static void main(String[] args)
    {
        int x=15,y=4;

        System.out.println("x y");
        while (x >= 0) {
            x = x - y;
            y = y + 3;
        }
        System.out.println(x + " " + y);
    }
}
```

```
(c) public class prob3_1c {
    public static void main(String[] args)
    {
        int s=8,t=0;

        while (s >= 4) {
            if (s > t)
                t += 3;
            if(s<=t)
                s += 2;
            System.out.println(s + " " + t);
            s -= 3;
            System.out.println(s + " " + t);
        }
    }
}
```

```

(d) public class prob3_1d {
    public static void main(String[] args)
    {
        int x=-3,y=1,z=2;

        while (x < z) {
            if (Math.abs(x + z) > Math.abs(y))
                System.out.println("yes");
            if (Math.abs(x) + Math.abs(z) > Math.abs(y))
                System.out.println("no");
            System.out.println("maybe");
            x++;
            z--;
        }
    }
}

(e) public class prob3_1e {
    public static void main(String[] args)
    {
        double b;
        int a=0;

        System.out.println("number    square root");
        while (a <= 6) {
            b = Math.sqrt(a);
            System.out.println(a + " " + b);
            a += 1;
        }
    }
}

(f) public class prob3_1f {
    public static void main(String[] args)
    {
        int c=1;
        double d;

        System.out.println("number    reciprocal    check ");
        while (c <= 8) {
            d = 1.0 / c;
            System.out.println(c + " " + d + " " + c*d);
            c += 1;
        }
    }
}

```

2. What is the value of ans at the end of the following **if-else** statement?

```

num1 = 12;
num2 = 14;
if (num1 + 3 < num2 - 2)
    ans = num1 - 9;
else
    ans = num1 + num2;

```

3. What is printed by the following section of code?

```
int num=9,x=12;

if (num >= x) {
    num = x;
    x = num - 2;
}
else {
    x = num;
    num += 3;
}
System.out.println("num = " + num + " and x = " + x);
```

4. What is wrong with the following loop?

```
x = 4;
while (x < 5);
{
    System.out.println(x);
    x++;
}
```

5. What is wrong with the following **if-else** statement? Note that it is not a simple syntax error--it *will* compile. Fix the error.

```
if (x = y)
    z = x + 1;
else
    z = x + 2;
```

6. For each of the following series of Java statements, try to describe in words what is accomplished (e.g., the larger of x and y is put into max , and then max is printed; or x is multiplied over and over by 2 until it is more than 30). For parts (a)-(c), (f), and (g), assume that x and y have already been given values. All variables have data type **int**.

```
(a)  if (x > y)
        max = x;
    else
        max = y;
    System.out.println(max);
```

```
(b)  if (x > y)
        ans = x;
    else
        ans = y;
    if (x == y)
        ans = 1
```

```
(c)  hold = 5;
    if (x == y)
        hold = x;
    else
        hold = y
```

```
(d)  x = 1;
    while (x < 100) {
        if (x < 50)
            x *= 2;
        x *= 2
    }
```

```
(e)  x = 0;
    while (x < 15) {
        x = x + 4;
        if (x == 8)
            System.out.println(x);
    }
```

```
(f)  x = 0;
    while (x < 15) {
        x = x + y;
        if (x < y)
            x = y;
    }
```

```
(g)  if (x == y)
        System.out.println(x);
    else
        System.out.println(y);
```

```
(h)  x = 0;
    while (x <= 6) {
        y = x + 4;
        if (y == 10)
            System.out.println("done");
        x += 1;
    }
```

7. The following program is preceded by a comment that describes the goal. Try to use the data to see if the segment works in all cases. If it does not, explain exactly what happens and show how to amend it so that it does always work.

```
// find the smallest integer whose
// square is greater than y
public class prob3_7 {
    public static void main(String[] args)
    {
        int number=0, test=0, y;

        while (test <= y) {
            number++;
            test = number + number;
        }
        System.out.println(number + " is the smallest");
    }
}
```

(try $y = 4$, $y = 5$, $y = 26$)

8. Assume that the user types in these values as requested:

10 -1 -1 -7 0 -1 4 14 4 6 13 5 9 0 42 42

Trace step by step the execution of this program. Assuming `myout` is redirected to a file, show exactly what is printed and where it appears (screen or file).

```
import java.util.Scanner;
import java.io.*;

public class prob3_8 {
    public static void main(String[] args)
    {
        int x,y;

        PrintWriter myout = new PrintWriter("myoutput.txt");

        Scanner kybd = new Scanner(System.in);

        System.out.print("Please enter an integer: ");
        x = kybd.nextInt();
        System.out.print("Please enter another integer: ");
        y = kybd.nextInt();
        while (x != y) {
            myout.print(x + " " + y);
            if (x > y)
                myout.println("\t" + x);
            else
                myout.println("\t" + y);

            System.out.print("Please enter two more integers: ");
            x = kybd.nextInt();
            y = kybd.nextInt();
        }
        myout.println("\nFinished");

        myout.flush();
        myout.close();
        kybd.close();
    }
}
```

9. Assume that the user enters the data values shown below on the four lines as needed (or that the data values come from a file). Trace step by step the values that are read into the variables a, b, c, and d. Show exactly what is printed by this program and where it is printed.

```
import java.util.Scanner;
import java.io.*;

public class prob3_8 {
    public static void main(String[] args)
    {
        double a,b,c;
        int    d;

        PrintWriter myout = new PrintWriter("myoutput.txt");

        Scanner kybd = new Scanner(System.in);

        System.out.print("Enter the first line of numbers: ");
        a = kybd.nextDouble();
        b = kybd.nextDouble();
        c = kybd.nextDouble();
        d = kybd.nextInt();
        myout.printf("%.2f %.2f %.2f %d\n", a, b, c, d);
        System.out.print("Enter the second line of numbers: ");
        a = kybd.nextDouble();
        System.out.print("Enter the third line of numbers: ");
        a = kybd.nextDouble();
        b = kybd.nextDouble();
        c = kybd.nextDouble();
        myout.printf("%.2f %.2f %.1f %d\n", a, b, c, d);
        System.out.print("Enter the fourth line of numbers: ");
        b = kybd.nextDouble();
        c = kybd.nextDouble();
        d = kybd.nextInt();
        myout.printf("%.2f %.2f %.1f %d\n", a, b, c, d);

        myout.flush();
        myout.close();
        kybd.close();
    }
}
1.35  2.1  -9.2  13
15
123.45  0  678
12.39  -5  5
```

10. For each number, determine what width & precision to use to print the value in the format shown.

- | | | |
|--------------|----------------|----------------|
| (a) 276.345 | (b) -8314.5678 | (c) 52.676 |
| (d) -0.00003 | (e) 0.000076 | (f) 1.14151617 |
| (g) 567.3 | (h) 493.888 | (i) -9.4 |
| (j) -19.6475 | (k) -203. | |