Activity 1, Section 1.2

Exercises

Exercise 3 : Suppose that `a` and `b` are `boolean` values. Show that the expression `(!(a && b) && (a || b)) || ((a && b) || !(a || b))` is equivalent to `true`.

Solution: The supposition is correct, I tested using the following code.

```
public class exe3{
   public static void main(String[] args) {
   boolean a, b, result;
   a = b = true;
   result = (!(a && b) && (a || b)) || ((a && b) || !(a || b));
   System.out.println(result);
   a = b = false;
   result = (!(a && b) && (a || b)) || ((a && b) || !(a || b));
   System.out.println(result);
   a = false;
        b = true;
   result = (!(a && b) && (a || b)) || ((a && b) || !(a || b));
   System.out.println(result);
   a = true;
        b = false;
   result = (!(a && b) && (a || b)) || ((a && b) || !(a || b));
   System.out.println(result);   }
}
```

The result of running this program is:

```
D:\JAVA\Activity1\Code>java exe3
true
true
true
true
```

Exercise 4 : Suppose that `a` and `b` are `int` values. Simplify the following expression: `(!(a < b) && !(a > b))`

Solution: (a == b)

Exercise 6 : Why does `10/3` give `3` and not `3.33333333`?

Solution: Since both 10 and 3 are integer literals, Java sees no need for type conversion and uses integer division. You should write `10.0/3.0` if you mean the numbers to be `double` literals. If you write `10/3.0` or `10.0/3`, Java does implicit conversion to get the same result.

Exercise 7 : What do each of the following print?

a. `System.out.println(2 + "bc");` prints: 2bc

b. `System.out.println(2 + 3 + "bc");` prints: 5bc

c. `System.out.println((2+3) + "bc");` prints: 5bc

d. `System.out.println("bc" + (2+3));` prints: bc5

e. `System.out.println("bc" + 2 + 3);` prints: bc23

Explain each outcome.

Solution:

a. Is a concatenation of 2 plus the string "bc"

b. Is a concatenation of the sum of 2 + 3 plus the string "bc"

c. The same that b) but the use of parenthesis avoid the type conversion validation

d. Using parenthesis, the compiler does the sum operation first and then the concatenation with the string "bc"

e. The compiler concatenates "bc" with 2, the result is a string, then concatenates with 3

Exercise 9 : What do each of the following print?

a. `System.out.println('b');`

b. `System.out.println('b' + 'c');`

c. `System.out.println((char) ('a' + 4));`

Explain each outcome.

Solution:

a. `Prints the character 'b'`

b. `Prints 197, the sum of the decimal ascii codes of b and c (98 + 99)`

c. `Prints the character 'e', because the compiler does the sum of the decimal ascii code of the character 'a'(97) + 4 resulting 101 which is the decimal ascci code for 'e'`

Exercise 10 : Suppose that a variable `a` is declared as `int a = 2147483647` (or equivalently, `Integer.MAX_VALUE`). What do each of the following print?

a. `System.out.println(a);`

b. `System.out.println(a + 1);`

c. `System.out.println(2 - a);`

d. `System.out.println(-2 - a);`

e. `System.out.println(2 * a);`

f. `System.out.println(4 * a);`

Explain each outcome.

Solution:

a. `Prints 2147483647, the value assigned to a`

b. `Prints -2147483648, becuase the sum of integer max value + 1 does the sum but gives negative result`

c. `Prints -2147483645, the result of 2 – 2147483647`

d. `Prints 2147483647, returns max int value due to the negative number`

e. `-2`

f. `-4`

Exercise 11 : Suppose that a variable `a` is declared as `double a = 3.14159`. What do each of the following print?

a. `System.out.println(a);`

b. `System.out.println(a + 1);`

c. `System.out.println(8 / (int) a);`

d. `System.out.println(8 / a);`

e. `System.out.println((int) (8 / a));`

Explain each outcome.

Solution:

a. Prints 3.14159, the value assigned to a

b. Prints 4.14159, the value of a + 1

c. Prints 2, the integer part of dividing 8 / 3 (integer part of a)

d. Prints 2.5464812403910124 the result of divide 8 / 3.14159 as a double

e. Prints 2, converts the double result of 8 / a to integer

Explain each outcome.

Exercise 20 : Suppose that a and b are int values. Simplify the following expression: `(!(a < b) && !(a > b))`

Solution: The result of running the program 3 times

```
D:\JAVA\Activity1\Code>javac D:\JAVA\Activity1\Code\SumOfTwoDice.java

D:\JAVA\Activity1\Code>java SumOfTwoDice

3

D:\JAVA\Activity1\Code>java SumOfTwoDice

2

D:\JAVA\Activity1\Code>java SumOfTwoDice

11
```

Exercise 29 : Write a program DayOfWeek.java that takes a date as input and prints the day of the week that date falls on. Your program should take three command-line arguments: m (month), d (day), and y (year). For m use 1 for January, 2 for February, and so forth. For output print 0 for Sunday, 1 for Monday, 2 for Tuesday, and so forth. Use the following formulas, for the Gregorian calendar:

Solution: The result of running the program is

```
D:\JAVA\Activity1\Code>java DayOfWeek 4 30 1995

0
```