Assignment 1 Bradley Justice

Question 1.

a, b = int, c = double

a/b\*c:

1. a/b = int result as both a & b are int’s
2. (a/b)\*c = double result, as the next operation is done with an int and a double, c, the result is type-promoted to a double.

2\*a+c:

1. 2\*a = int result as both variables are of type int
2. (2\*a)+c = double result, the type gets promoted to double as the computer automatically type promotes when an operation is done using both types.

7.8\*(a\*b)+c:

1. first variable 7.8, is a double
2. (a\*b) = int type, as both variables are ints
3. 7.8\*(a\*b) = double result, type promotion applied to (a\*b) as it interacts with the double
4. 7.8\*(a\*b)+c = double result, type has already been promoted to double so it stays that way

2\*a+2.0\*a:

1. 2\*a = int result, both variables are of type int
2. 2\*a + 2.0 = double result, type promotion as 2.0 is automatically assigned type double as it has a decimal.
3. 2\*a+2.0\*a = double result

Question 2.

Int a = 12; declared as int by code; evaluates RHS as int, then as LHS confirms int type, no additional work done.

Int b = 5; declared as int by code; evaluates RHS as int, then, as LHS confirms int type, no additional work done.

Int c = a/b; declared as int by code; evaluates RHS 12 = int / 5 = int, so there is no type promotion, and 5 fits into 12 twice with 2 remaining, remainder is ignored, result is rounded down to nearest int, 2, then LHS confirms int type, no additional work is done.

Float d = a \* c + 12f; declared as float by code (end result will be float or converted to float); evaluates RHS to 12\*2+12.0; 12\*2 = 24, then, 24+12.0 = 36.0; type is promoted from int to float as 12.0 is assigned type float from the ‘f’ in the code, then, as 36.0 is already a float, when it is assigned to d, no additional work is needed for casting/converting as evaluation already does this.

Double e = 7.8f \* (d-b) + c; declared as double by code; evaluates RHS to 7.8f\*(36.0-5)+2; comes out to be 243.8, but of type float, as that is the largest type on the RHS, then, as the LHS declares is to be of type double, the answer is type promoted to a double.

Question 3.

Explain the output of the program and why it is what it is:

The output and relevant bits of the program are:

Int a = 12

Int b = 5

Int z = a+b

12 (print a)

5 (print b)

17 (print z)

a++

b++

13 (print a)

6 (print b)

17 (print z)

At the beginning of the program, a, b & z are declared and initialized as 12, 5, and 17 respectively. Z is computed at line 3 to be equal to 17, 12+5. The computer is asked to print out a, b and z, which it does so as 12, 5, and 17. Then, in the lines a++ and b++, variables a and b are declared to be one greater than before; 13 and 6 respectively. At this point, the computer is asked to print a, b and z again, and does so as 13, 6, and 17. The z variable does not change as it has already been computed in line 3 as 17, before a and b are changed. Furthermore, z is not re-evaluated, so it just holds the same value computed earlier.