Programming Logic – Beginning

Additional Processing Examples (quiz review)

1. Declare a variable to hold the number of Legos in a bucket.
```
int legos;            //Integer because can't have fractional Legos
```

2. Assign the value 1,065 to that variable
```
legos = 1065;         //Note commas not allowed in code
```

3. Declare a variable to hold your student ID
```
string studentId;     //Don't do math with it
```

4. Assign the value 12345678 to that variable
```
studentId = "12345678";
```

5. Declare a variable to hold the start of Spring Break
```
DateTime springBreakStart; //Variable description here a good idea
```

6. Assign March 17, 2014 to that variable
```
springBreakStart = new DateTime(2014, 3, 17); //Year, month, day
```

7. Write a C# statement to determine the end of Spring Break (7 days later)
```
'Declaration not requested
springBreakEnd = springBreakStart.AddDays(7); //Can't use +
```

8. Declare a variable to hold the weight of the bucket of Legos.
```
double legoWeight ;         //Next statement uses double
```

9. Assign value 5¼ to that variable.
```
legoWeight = 5.25;
```

10. Write VB statement to increase the weight by $1\frac{1}{3}$
```
legoWeight = legoWeight + 1.333;       //OK
legoWeight = legoWeight + 1 + 1 / 3.0;  //Better, retains more decimals
legoWeight += 1.333;
legoWeight += 1 + 1 / 3.0;             //Note: 3.0 (required)
                                       //Any of these are acceptable
```

```
int intVar1 = 10;
int intVar2 = 5;
int intVar3 = 3;
double dblResult;
int intResult;

intResult = intVar1 % intVar2;                          intResult =
intResult = intVar2 % intVar1;                          intResult =
intResult = intVar2 / intVar1;                          intResult =
dblResult = intVar2 / intVar1;                          dblResult =
dblResult = (double) intVar2 / intVar1;                 dblResult =

intResult = intVar1 + intVar2 / intVar3;                intResult =
dblResult = (double)intVar1 / intVar2 / intVar2;        dblResult =
```

```
intResult = intVar1 % intVar2;    10 mod 5              intResult = 0
intResult = intVar2 % intVar1;     5 mod 10             intResult = 5
intResult = intVar2 / intVar1;     5 / 10               intResult = 0
     'Integer division

dblResult = intVar2 / intVar1;     5 / 10               intResult = 0.0
     'Still integer division, assigned to double

dblResult = (double) intVar2 / intVar1;     5 / 10      dblResult = 0.5

intResult = intVar1 + intVar2 / intVar3;    10 + 5 / 3  intResult = 11

dblResult = (double)intVar1 / intVar2 / intVar2;        dblResult = 0.4
                  10 / 5 / 5
                     2   / 5
```

```
double a, b, c, d;                          //Not recommended but legal

a = 0;
b = 15;
c = 5;

a = a + 10;    //a becomes 10.0
a += 10;       //a becomes 20.0

d = a * b / c;  //d becomes 60.0
    20 * 15 / 5
      300   / 5
b = Math.Pow(a, 2);       //b becomes 400.0 (same as 20 * 20)
c = c % b;     //c becomes 5.0    (5 mod 400)
```

Final values:

a = 20.0
b = 400.0
c = 5.0
d = 60.0

The surface area of a cube is defined as: area = $6a^2$ (Where a is the length of each side)
Write the equivalent C# formula

```
area = 6 * Math.Pow(a,2);      //Best
area = 6 * a * a;              //OK
    'OK to rename a → length
```

The surface area of a rectangular box is defined as: area = 2lh + 2hw + 2lw
Where h=height l=length w=width
Write the equivalent C# formula

```
area = 2*l*h + 2*h*w + 2*l*w;  //( ) optional, * are not
  //OK to rename l, h, w → length, height, width
```