

Name _____ Period _____

1. Write a driver class called "NumericVariables". In the driver class, (2 points)

Indicate each part below in your code with appropriate comments. For example, for part a type "//Part a". (1 point)

- a) Declare and initialize an int a1. Write a single line of code that will print to the console the value of a1 decremented by 1. (2 points)
- b) Declare and initialize an integer b1. Declare and initialize an integer b2. Then write a single line of code that uses the compound operator, -=, to subtract b2 – 30 from the value of b1 and store the result back in b1. (2 points)
- c) Initialize an integer variable c1 and another integer c2 to a value smaller than c1. Write a program that prints the remainder that results when these two numbers are divided. (2 points)
- d) Write code that will create a constant variable D1 that is equal 2.718. (1 point)
- e) Declare and initialize a double variable e1. Declare and initialize an int variable e2. Declare an int variable e3, and assign the value of e1 divided by e2 to e3. (2 points)

```
public class NumericVariables{  
  
    public static void main(String args[]){  
        //part a  
        int a1 = 5;  
        System.out.println(--a1);  
  
        //part b  
        int b1 = 2, int b2 = 3;  
        b1 +=(b2 - 30);  
  
        //part c  
        int c1 = 5, c2 = 2;  
        System.out.println(c1%c2);  
  
        //part d  
        final double D1 = 2.718;  
  
        //part e  
        double e1 = 5;  
        int e2 = 10;  
        int e3 = e1/e2;  
  
    }  
}
```

2. For each of the following code segments, indicate the output. (1 point each)

(a)

```
int a = 5;  
int b = 11;  
b*=a;  
System.out.println(b + 1);
```

56

(b)

```
double m = 41;  
int j = 5;  
m = m/j;  
System.out.println(m);
```

8.2

The following code applies to (c) thru (h)

```
int dividend = 6, divisor = 2, quotient = 0, remainder = 0;
int dividend2 = 5, divisor2 = 3, quotient2 = 0, remainder2 = 0;
```

```
quotient = dividend/divisor;
remainder = dividend % divisor;
quotient2 = dividend2/divisor2;
remainder2 = dividend2 % divisor2;
```

```
(c) System.out.println(quotient);      3
```

(d) `System.out.println(remainder);` **0**

```
(e) System.out.println(quotient2);    1
```

(f) `System.out.println(remainder2);` **2**

```
(g) System.out.println(quotient += quotient2); 4
```

(h) `System.out.println(++remainder);` **1**

/8

3. Consider the code below. Predict the result of each of the following numeric operations

```
double d1 = 37.9;
double d2 = 1004.128;
int i1 = 12;
int i2 = 18;
```

(a) $57.2 * (i1 / i2) + 1$

1.0

(b) $57.2 * ((double)i1 / i2) + 1$

39.13333333333333

(c) $15 - i1 * (d1 * 3) + 4$

-1345.3999999999999

(d) $15 - i1 * ((int)d1 * 3) + 4$

-1313

/4

4. Write code that could be used to write a number backwards. Your code should work for any number with 4 digits. Consider the int data type below,

```
int number = 1234;
```

When your code is ran, "4321" should print to the console.

Below are more examples,

int data type	result
int n1 = 3455;	5543
int n2 = 8767;	7678
int n3 = 2468;	8642

```
int n = 1234;
int ones = n%10;
int tens = (n/10)%10;
int hundreds = (n/100)%10;
int thousands = (n/1000)%10;
String reversed = "" + ones + tens + hundreds + thousands;
System.out.println(reversed);
```

/3