Functions Gone Wild 2020

This problem will be worth 16 points. Each successfully implemented function will earn you a point. There is the student tester and one test for each of the 10 functions for a total of 11 possible points. Additional bonus points will be awarded for correctly implementing functions 1-3, functions 4 - 6, functions 7 and 8, and functions 9 and 10. A final bonus point will be awarded for correctly implementing all 10 functions.

This problem will ask you to implement the following functions. One function is defined recursively. On some problems I will give you functions from the java.lang.Math class with a brief description, and sometimes you just need to implement the function. In all the following functions you do **NOT** need worry about domain issues. That is, you may assume that all test data will not cause any exceptions to be thrown. In total, there are ten different functions in this problem. All methods which return a `double` must return a value 'close enough' (less than some delta > 0) to the correct answer to be considered correct.

You should use:

• `Math.max(a,b)` for max(a, b).	• `Math.min(a, b)` for min(a, b)
• `Math.abs(x)` for |x|.	• `Math.cos(a)` for cos(a)
• `Math.sqrt(x)` for $\sqrt{x}$	• `Math.sin(a)` for sin(a)
• `Math.pow(x, 1.0/n)` for $\sqrt[n]{x}$	• `Math.tan(a)` for tan(a)
• `Math.log(a)` for $\ln(a)$	• `Math.log10(a)` for $\log_{10}(a)$ or $\log(a)$
• `Math.ceil(a)` for $\lceil a \rceil$.	• `Math.floor(a)` for $\lfloor a \rfloor$.

note:

- All trig functions are in radians.
- `Math.abs(int a)` returns an `int` and `Math.abs(double a)` returns a `double`
- `Math.max(int a, int b)` and `Math.min(int a, int b)` returns an `int`.
- `Math.max(double a, double b)` and `Math.min(double a, double b)` returns a `double`.
- Return type of all other methods is double.
- $\lfloor x \rfloor$ is the largest (Closes to positive infinity) double value smaller than or equal to the argument, `x`, and is equal to a mathematical integer.. For example, $\lfloor 2.9 \rfloor = 2.0$ and $\lfloor -14.3 \rfloor = -15.0$.
- $\lceil x \rceil$ is the smallest (Closes to negative infinity) double value that is greater than or equal to the argument, `x`, and is equal to a mathematical integer. For example, $\lceil 1.1 \rceil$=2.0 and $\lceil -24.9 \rceil = -24$.

- Use the following constant for pi (π)

`Math.PI`	The `double` value that is closer than any other to *pi*, the ratio of the circumference of a circle to its diameter.

- Use the following constant for e

`Math.E`	The `double` value that is closer than any other to *e*, the base of the natural logarithms.

Functions Gone Wild 2020

1. A sequence $a_1, a_2, a_3, \ldots$ is defined by the equation:
   $a_1 + a_2 + a_3 + \ldots + a_n = n^3$ or $a_n = n^3 - (a_1 + a_2 + \ldots + a_{n-1})$

   This implies: $a_1 = 1^3 = 1$,
   $a_2 = 7$ since $a_2 = 2^3 - a_1 = 8 - 1 = 7$,
   $a_3 = 19$ since $a_3 = 3^3 - (a_2 + a_1) = 27 - 8 - 1 = 19$,
   etc.

   That is, returns a_n that makes $a_1 + a_2 + a_3 + \ldots + a_n = n^3$ true.

Use following function heading:

```
public static int f1(int n)
```

```
Test data:  f1(1) returns 1  : 1^3 = 1
            f1(2) returns 7  : 2^3 = 1+ a2 = 8, a2 = 8 - 1 = 7
            f1(3) returns 19 : 3^3 = 1+7+ a3= 27, a3 = 27 - 8 = 19
            f1(4) returns 37 : 4^3 = 1+7+19+ a4= 64, a4 = 64 - 27 = 37
            f1(5) returns 61 : 5^3 = 1+7+19+37+ a5= 125, a5 = 125 - 64 = 61
            f1(6) returns 91 : 6^3 = 1+7+19+37+61+ a6= 216, a6=216 - 125 = 91
```

2. Find the number of possible order pairs of digits (a, b) such that the `String` parameter `str` (of the form #...#a#...#b#...# where #...# represents 0 or more consecutive digits: 0, 1, 2, 3, ,,,, 8, 9) is divisible by `int` parameter `div`.
   You may assume that when a and b are both replaced with 9, the new value will be a legal `int` value.
   You may assume all values will be positive `int`s.
   You may assume a will be located before b. That is, a will always be in the larger place value.
   You may assume `div != 0`
   a may be leading digit, and 0 replacing a is acceptable.

For example, `f2("1ab3", 17)` returns 6 because only 6 numbers (1003, 1173, 1343, 1513, 1683, 1853) are divisible by 17.

Use following function heading:

```
public static int f2(String str, int div)
```

```
Test data:  f2("1ab3", 17) returns 6
            f2("1a1b1", 3) returns 34
            f2("ab", 2) returns 50
            f2("3a123b", 5) returns 20
```

Functions Gone Wild 2020

3. Implement the following integer recursive function. All calculations shall be computed using Integer math.

$$f3(n) = \begin{cases} f3\left(\frac{2n^2+21}{5(n-25)}\right) + \frac{n}{3} & n \geq 200, n \text{ is even} \\ f3\left(\frac{12n+27}{11n} + \frac{n}{5}\right) + 25 & n \geq 200, n \text{ is odd} \\ f3(n \,\%\, 20) + n & 50 < n < 200 \\ (n-2)^2 & otherwise \end{cases}$$

Use following function heading:

```
public static int f3(int n)
```

```
Test data:  f3(20) returns 324  = (20 - 2)^2
            f3(135) returns 304 = f3(15) + 135
                                = 169 + 135
            f3(501) returns 127 = f3(101) + 25
                                = f3(1) + 101 + 25
                                = 1 + 101 + 25
            f3(500) returns 500 = f3(210) + 166
                                = f3(195) + 70 + 166
                                = 169 + 95 + 70 + 166
```

4. Implement the following function

Consider the mathematical notation $\sum_{i=m}^{n}(someFunction)$ used to represent the summation of many similar terms.

The notation $\sum_{i=m}^{n}(h(i))$ is defined as: $h(m)+h(m+1)+h(m+2)+\ldots+h(n)$

The subscript gives the symbol for an index variable, i. Here, i represents the index of summation; m is the lower bound of summation, and n is the upper bound of summation. In this case, i = m under the summation symbol means that the index i starts equal to m. Successive values of i are found by adding 1 to the previous value of i, continuing up to and including when i equals n. An example: $\sum_{k=2}^{6} k^2 = 2^2+3^2+4^2+5^2+6^2 = 90$.

Your task in this problem is to implement the following function.

Note – all calculations are to be completed using integer math

$$f4(a,b,c) = \sum_{i=\max(a-b,1+|b-c|)}^{\min(ab+c,bc+a)} \left(\left[\frac{3a(a+b)}{1+a*|b-c|} \right] i^2 + \left[\frac{max(b,c)*min(ab,|a-c|)}{max(1,max(a,b) - min(b,c))} \right] i \right)$$

Which is equivalent to the following:

Functions Gone Wild 2020

$$\left[\frac{3a(a+b)}{1+a*|b-c|}\right]\left(max(a-b,1+\lfloor b-c\rfloor)\right)^2+\left[\frac{max(b,c)*min(ab,|a-c|)}{max(1\,,max(a,b)\ -\ min(b,c)\ )}\right]max(a-b,1+\lfloor b-c\rfloor)$$

$$+\left[\frac{3a(a+b)}{1+a*|b-c|}\right]\left(1+\max(a-b,1+\lfloor b-c\rfloor)\right)^2$$
$$+\left[\frac{\max(b,c)*\min(ab,|a-c|)}{\max(1,\max(a,b)-\min(b,c)\ )}\right][1+\max(a-b,1+\lfloor b-c\rfloor)]$$

$$+\left[\frac{3a(a+b)}{1+a*|b-c|}\right]\left(2+\max(a-b,1+\lfloor b-c\rfloor)\right)^2$$
$$+\left[\frac{\max(b,c)*\min(ab,|a-c|)}{\max(1,\max(a,b)-\min(b,c)\ )}\right][2+\max(a-b,1+\lfloor b-c\rfloor)]$$

+.....+

$$+\left[\frac{3a(a+b)}{1+a*|b-c|}\right]\left(\min(ab+c,bc+a)\right)^2$$
$$+\left[\frac{\max(b,c)*\min(ab,|a-c|)}{\max(1\,,\max(a,b)-\ \min(b,c)\ )}\right][\min(ab+c,bc+a)]$$

Special Note: $if\ \min(ab+c,bc+a) <\ max\ (a-b,1+|b-c|)$, return 0.

Use following function heading:

```
public static int f4(int a, int b, intc)
```

```
Test data:
   f4(2, 2, 1)
        returns = f(2) + f(3) + f(4) = 36 + 78 +136 = 250

   f4(1, 2, 5)
        returns = f(4) + f(5) + f(6) + f(7) = 72 + 100 + 132 + 168 = 400
```

Functions Gone Wild 2020

5. What is the smallest `int`, when adjoined/concatenated to the end of the first parameter (num) is divisible by the second parameter, `div`.
   You may assume `both num > 0 and div > 0`.
   Remember, you are to return an `int`, which do not contain leading zeros.

Use following function heading:

-

```
public static int f5(int num, int div)
```

```
Test data:  f5(523, 3)   returns 2,   5232%3 == 0
                                      (5230 & 5231 are not divisible by 3)
            f5(154, 11)  returns 0,   1540 % 11 == 0
            f5(1, 35)    returns 40,  cannot return 05
                                      next value divisible by 35 is 140
            f5(98, 11)   returns 12,  [980 .. 989] % 11 != 0
                                      cannot return 01, 9812 % 11 == 0
```

6. Implement the following function

$$f6(x,y,z) = \begin{cases} 10 * log(\pi^{x+z}) & sin(x) > cos\left(\frac{z}{y}\right) \\ 25 * ln(y^{e+z}) & sin(x) \leq cos\left(\frac{z}{y}\right) \end{cases}$$

Use following function heading:

```
public static double f6(double x, double y, double z)
```

```
Test data:  f6(4*Math.PI/5., 3., 3*Math.PI/2)  returns 35.922375
            f6(4*Math.PI/5., 2., Math.PI/3)  returns  65.250785
```

Functions Gone Wild 2020

7. Implement a function which returns a String containing all lower letters not contained in the String parameter. The String being returned must be in alphabetical order.

Additional Information:

- The String parameter will only contain spaces, Upper Case letters, Lower Case letters, numbers, periods (.) and question marks (?).
- Since only lower case letters are being returned, all other characters (Spaces, Upper Case letters, numbers, periods and question marks) should be ignored!

Use following function heading:

```
public static String f7(String phr1)
```

```
Test data:  f7("Computer Science")  returns "abdfghjklqsvwxyz"
            f7("The quick brown fox jumps over lazy dog.")  returns "t"
            f7("Programming Contest 2020")  returns "bcdfhjklpquvwxyz"
```

8 Implement the following function

This method will scramble the `String` phrase by replacing each individual letter using a simple letter substitution scheme. The scramble is performed by replacing the letters from A to J with the letter advanced five letters (A s replaced with F, B is replaced by G, …. and J is replaced by O.

Similarly the letters from K to Z are replaced according to the following scheme: Z is replaced by P, Y is replaced Q, X is replaced by R, …, P is replaced by Z, Q is replaced by A, …, and K is replaced by E.

You may assume the String contains only UPPER case letters and spaces (spaces remain unchanged).

Use following function heading:

```
public static String f8(String phrase)
```

```
Test data:   f8("ABCDEFGHIJ") returns "FGHIJKLMNO"
             f8("ZYXWVUTSRQPONMLK") returns "PQRSTUVWXYZABCDE"
             f8("GOT IT") returns "LAV NV"
```

Functions Gone Wild 2020

9. Implement the following function

$$f9(x,y,z) = \begin{cases} true & x = false & y = false & z = false \\ false & x = false & y = false & z = true \\ false & x = false & y = true & z = false \\ true & x = false & y = true & z = true \\ true & x = true & y = false & z = false \\ false & x = true & y = false & z = true \\ true & x = true & y = true & z = false \\ false & x = true & y = true & z = true \end{cases}$$

Use following function heading:

```
public static boolean f9(boolean x, boolean y, boolean z)
```

```
Test data:  f9(false, false, false) returns true
```

10. Implement the following function

$$f9(x,y,z) = \begin{cases} false & j = false & k = false & m = false & n = false \\ true & j = false & k = false & m = false & n = true \\ true & j = false & k = false & m = true & n = false \\ false & j = false & k = false & m = true & n = true \\ false & j = false & k = true & m = false & n = false \\ true & j = false & k = true & m = false & n = true \\ true & j = false & k = true & m = true & n = false \\ true & j = false & k = true & m = true & n = true \\ true & j = true & k = false & m = false & n = false \\ false & j = true & k = false & m = false & n = true \\ false & j = true & k = false & m = true & n = false \\ false & j = true & k = false & m = true & n = true \\ true & j = true & k = true & m = false & n = false \\ true & j = true & k = true & m = false & n = true \\ false & j = true & k = true & m = true & n = false \\ false & j = true & k = true & m = true & n = true \end{cases}$$

Use following function heading:

```
public static boolean f10(boolean j, boolean k, boolean m, boolean n)
```

```
Test data:  f10(false, false, false, false) returns false
```