CSCI 1130
Assignment 5
3/18/19

The references below apply to the "Course Notes". See link on our D2L homepage. See D2L Submission Folder for Due Date.

This assignment dives into data types, operators and basic logic.

Read the following chapters in Part III:

- Chapter 1 -- Primitive Data Types and Operators
- Chapter 2 -- Basic Logic

General guidelines:

- All instance variables should be private. All methods specified in this write-up should be public. Other methods may be public, private, protected as you wish.
- Code your classes rich with “helper methods”
- Reuse code including your own (try not to reinvent code)
- Use your “README.TXT / Samples Approach”

A. Design and Code BrilliantNumber

Here you'll teach a basic simple number to be brilliant.

Call the class “BrilliantNumber”.

Instance Variables:

Name	Type	Notes
num	int	You may assume num is positive natural number (>=1). In other words, you need not concern yourself with num <= 0

Constructors:

Name	Parameters	Notes
BrilliantNumber	int newNum	Set ivar

Instance Methods : (all methods below reference our ivar “num” unless otherwise noted). **Also see** more information below this table.

Name[(See Note 1)]	Parameters	Return Type	Description[1]
getNum	None	int	Return ivar
setNum	int newNum	None	Set ivar
isEven	-	boolean	Return true if we are an even number
isOdd	-	boolean	Return true if we are an odd number
isSquare	-	boolean	Return true if we are a square number, 1, 4, 9, 16, etc are square numbers
isCube	-	boolean	Similar to isSquare except if we are a cube number
isSquareAndCube	-	boolean	Return true if we are both a square and cube
isDivisibleBy	int divisor	boolean	Return true if we are divisible by method param[(See Note 2)].
isPalindrome	-	boolean	Return true if “num” is a palindromic number
squared	-	int	Return square of self, e.g. 3*3=9, 11*11=121, etc
cubed	-	int	Return cube of self, 3*3*3=27, 10*10*10=1000, etc
roundedRatio	int divisor	int	Return rounded ratio[(See Note 2).] e.g. 2/1=2, 3/2=2, 5/2=3, 10/3=3.
remainder	int divisor	int	Return remainder of our number divided by the method param[(See Note 2).]
toString	-	String	Return nice string describing the specific object

Note (1): See “More Details” below for additional information
Note (2): For all the method parameters named “divisor”, assume that the divisor is equal or greater than one.

More Details:

Reference Table 1 below for related useful Java functions.

isSquare():

First devise algorithm. Example may help.
For num=361,
square root of 361 is 19.0
rounded is 19
19*19 = 361
361 is equal to “num”, so yes 361 is indeed a square number

isCube()

Similar to isSquare except we would use cube roots and cubes.

isPalindrome()

A palindrome reads the same forward and backward --
e.g. 5, 33, 333, 1001, 95259 are palindromes. See “Reversing a String” in Table 1 below.

remainder()

See Table 2 below.

Table 1 – Useful Java Functions

Function Type	Sample
Square Root	`Math.sqrt(x)`
Cube Root	`Math.pow(x, 1.0/3.0)`

Reversing a String	`String s = "101";`
	`StringBuilder sb = new StringBuilder();`
	`sb.append(s);`
	`sb.reverse();`
	`String reversedString = sb.toString();`

Converting an int to a String	`String s1 = "" + 10;`
	`//or`
	`String s2 = String.valueOf(10);`

Rounding	//round a double to an int
	double d = 5.3;
	int a = (int)Math.round(d)

Dividing ints to get a double	`int a=10, b=3;` `double x = ((double)a)/b;`

Table 2 – Remainder Samples

num	divisor	remainder
1	2	1
2	2	0
3	2	1
4	2	0
5	2	1
10	3	1
10	4	2
10	5	0
10	6	4
10	7	3
10	99	10
11	99	11
12	999999	12
13	9028348922	13

B. Problems in “Chapter 2 -- Basic Logic”

Go to:

- Course Notes
- Chapter 2 -- Basic Logic
- Go to "Problems" (bottom of page)
- Do problems #1, #2 and #4.
- Extra Credit: Problem #3 (10 Possible Points)

Submit Instructions

- Include your JAVA files in the ZIP.
- Note that all the JAVA filenames for this assignment are pre-defined. E.g. use the specified name “BrilliantNumber.java” and not “IntelligentNumber.java” the reason being is that grading is systematic.
- You may **yes** submit extra files in the ZIP except as mentioned below. (extra files will be ignored)

- Please do not submit two JAVA files with the same name (in different directories). E.g. do not submit two BrilliantNumber.java files, as the grader will not know which is preferred
- An improper submit makes grading much more difficult and could receive up to -5 points
- Name your ZIP file per these instructions in the course notes here:
  "General Guidance -- Course Instructions -- ZIP File for Submit"
- Submit your ZIP file in the D2L proper assignment folder

Assignment Scoring

Submit Schedule	Points	Note
If Submitted By Due Date	100	Due Date is given in D2L
If Submitted By **Late** Due Date	70	Late Due Date is one week after due date
If Submitted **after** "Late Due Date"	0	