



CSE215L: Programming Language II Lab

Chapter 1-6 + Recursion

Weight: 10% of the course

Question 1 [25]

There are two variants of the International Standard Book Number (ISBN): ISBN-10 and ISBN-13. You are to write a program that will help users verify whether an ISBN number is valid or not. Start your program by asking the user whether they want to enter ISBN-10 or ISBN-13 number. **You must use Enums in this part.** Once you know the type of ISBN the user wants to enter, prompt the user to enter the ISBN number.

Write a single static method that takes the ISBN number and the Enum provided by the user as input parameters and returns a boolean i.e. the validity of the provided ISBN. The following tables shows how to validate ISBN numbers:

ISBN-10	ISBN-13
a. Must be strictly 10 digits long. $d_1d_2d_3d_4d_5d_6d_7d_8d_9d_{10}$. The last digit can be the character 'X' as well.	a. Must be strictly 13 digits long. $d_1d_2d_3d_4d_5d_6d_7d_8d_9d_{10}d_{11}d_{12}d_{13}$
b. The last digit d_{10} is a checksum. It is calculated using the following formula: $(d_1 \times 1 + d_2 \times 2 + d_3 \times 3 + d_4 \times 4 + d_5 \times 5 + d_6 \times 6 + d_7 \times 7 + d_8 \times 8 + d_9 \times 9) \% 11$ The % at the end indicates a remainder operation. A convention is that if the value of the checksum is 10, it is denoted using the character 'X'.	b. The validity of an ISBN-13 number is calculated by the following formula: $(d_1 \times 1 + d_2 \times 3 + d_3 \times 1 + d_4 \times 3 + d_5 \times 1 + d_6 \times 3 + d_7 \times 1 + d_8 \times 3 + d_9 \times 1 + d_{10} \times 3 + d_{11} \times 1 + d_{12} \times 3 + d_{13} \times 1) \% 10$ The % at the end indicates a remainder operation.
c. If the checksum is not between 0 and 10, the ISBN is not valid.	c. If the remainder obtained from the provided operation is not 0, the ISBN is not valid.

In your main method, **take at least three ISBN numbers of different types** as user input using the Scanner class and print out whether they are valid or not.

Sample (format your output like this):

```
Please enter the ISBN type: ISBN_10
Please enter the ISBN-10 number: 013031997X
013031997X is a valid ISBN-10 number
```

```
Please enter the ISBN type: ISBN_13
Please enter the ISBN-13 number: 9780716703440
9780716703440 is a valid ISBN-13 number
```



CSE215L: Programming Language II Lab

Chapter 1-6 + Recursion

Weight: 10% of the course

Question 2 [25]

In Wyoming Limited, the starting monthly salary of an employee is BDT 20,000. Every six months, the monthly salary of the employee is incremented by 12.5% of the current monthly salary. As such, after the first year, the monthly salary of the employee will be BDT 25,313. Write a program that will calculate the monthly salary of the employee after 12 years.

Also calculate and display the number of years (rounded) the employee will have to save up to buy a car that costs BDT 2,500,000, assuming that he saves 45% of his monthly salary every month.

Output format:

Monthly salary after 1 year: BDT 25313

Monthly salary after 2 years: BDT 32037

Monthly salary after 3 years: BDT X1

Monthly salary after 4 years: BDT X2

...

Monthly salary after 12 years: BDT X10

Total number of years needed to buy a car worth BDT 2,500,000 assuming 45% of monthly salary is saved: Y years.



CSE215L: Programming Language II Lab

Chapter 1-6 + Recursion

Weight: 10% of the course

Question 3 [25]

Write the methods with the following signatures:

```
public static double computeAreaOfNonagon(double lengthOfSide)
public static double computeAreaOfDecagon(double lengthOfSide)
```

Area of a Nonagon with a side length of a is:

$$A = \frac{9}{4} a^2 \cot\left(\frac{180^\circ}{9}\right)$$

Area of a Decagon with a side length of a is:

$$A = \frac{5}{2} a^2 \sqrt{5 + 2\sqrt{5}}$$

Now, in your main method, use a loop to count from 10 to 50 (inclusive). Let's call the loop variable `sideLength`. **In each iteration, `sideLength` should increment by square root of 10.** Inside the loop, print out the area of a nonagon and decagon with side length of the current value of the loop variable.

Output format:

Area of a Nonagon with a side length of 10 is 618.18242, and that of a Decagon is 769.42088

Area of a Nonagon with a side length of 13.16 is 1070.60293, and that of a Decagon is 1332.52617

Area of a Nonagon with a side length of 16.32 is 1646.48189, and that of a Decagon is 2049.29405

... (rest of the values upto 50).



CSE215L: Programming Language II Lab

Chapter 1-6 + Recursion

Weight: 10% of the course

Question 4 [25]

Write a recursive method with the following signature:

```
public static double recursiveLogSeriesSum(int n)
```

This method takes a positive integer n and returns the sum of the following series:

$$y = 1 + \frac{1}{\log(2)} - \frac{1}{\log(3)} + \frac{1}{\log(4)} - \frac{1}{\log(5)} + \dots \pm \frac{1}{\log(n)}$$

Implement an iterative solution to the same problem in a different method with the signature:

```
public static double iterativeLogSeriesSum(int n)
```

In your main method, call the recursive method with three values (**use of Scanner class is mandatory**). Also call the iterative method with the same three values to verify that both your solutions are working correctly.

Example:

For $n = 5$, you need to compute $1 + \frac{1}{\log(2)} - \frac{1}{\log(3)} + \frac{1}{\log(4)} - \frac{1}{\log(5)}$ and the sum is 1.6324684001469958

Note: `Math.log()` in Java actually computes the log of base e. `Math.log10()` computes the log of base 10. The above sample output has been derived **using `Math.log()`** method.

Output format:

Using recursive method:

n = 5 yields 1.6324684001469958

n = 8 yields 2.15757903129148

n = 2 yields 2.4426950408889634

Using iterative method:

n = 5 yields 1.6324684001469958

n = 8 yields 2.15757903129148

n = 2 yields 2.4426950408889634