

Module 5 - Methods

CMPT220L

Due on Oct 15, 2020 by 11:59 PM

Points: 100

Problems

1. (*Decimal to binary*) Write a method that parses a decimal number into a binary number as a string. The method header is:

```
public static String decimalToBinary(int value)
```

Write a test program that prompts the user to enter a decimal integer value and displays the corresponding binary value.

Here is a sample run:

```
Enter an integer: 1451
The binary value is 10110101011
```

2. (*Capitalize first letter of each word*) Write the following method that returns a new string in which the first letter in each word is capitalized.

```
public static void title(String s)
```

Write a test program that prompts the user to enter a string and invokes this method, and displays the return value from this method. Here is a sample run:

```
Enter a string: london england 2015
The new string is: London England 2015
```

Note that words may be separated by multiple blank spaces

3. (*Hours, minutes, and seconds*) Write a method that returns a string in the form of hour:minute:second for a given total seconds using the following header:

```
public static String format(long seconds)
```

Here is a sample run:

```
Enter total seconds: 342324
The hours, minutes, and seconds for total seconds 342324 is 23:05:24
```

Note that a zero is padded to hour, minute, and second if any of these values is a single digit.

4. (*Pascal triangle*) Write a program that displays a Pascal triangle. The program prompts the user to enter the number of rows and displays the triangle. Here is a sample run:

Enter the number of rows: 8

```

      1
     1 1
    1 2 1
   1 3 3 1
  1 4 6 4 1
 1 5 10 10 5 1
1 6 15 20 15 6 1
1 7 21 35 35 21 7 1
1 8 28 56 70 56 28 8 1
```

For information on Pascal triangle, see <http://www.mathsisfun.com/pascals-triangle.html>. You need to write a method that computes $C(m, n) = \frac{m!}{(m-n)!n!}$

Submission

Make sure you create one Java file per project. Place your `.java` files under the corresponding folder in your local copy of the GitHub repository, commit and push it to the remote repository. Make sure that the professor has access to the repository (`jfac65-marist`).

```
cmpt220lastname\
  hw\
    5\
      Problem1.java
      Problem2.java
      Problem3.java
      Problem4.java
```