CodeLM 2016

Intermediate Division

School Name

Your password is: _____

Wi-Fi SSID: Programming

Wi-Fi Key: comp1234

Rules: You will have two and a half hours to complete your division's seven questions. All questions must be submitted through the website codeLM.com and have a runtime of less than 10 seconds. You will submit your code by copying and pasting the code from your IDE to each problem's textbox on the website and then hitting submit.

All method headers must exactly match the method header that is provided in the starter code and submission textbox. You may also submit other methods that your code may use. You should not submit a main method.

Problems 1 and 2 are each worth 2 points; Problems 3 and 4 are each worth 3 points; Problems 5 and 6 are each worth 4 points. In order to receive credit for these questions your code must produce the correct results for all test data. The test data will not be the same as the sample data that is provided with each question in this packet, so be sure to thoroughly test your code before you submit it. Incorrect submissions that compile will result in the loss of one point. If your submission does not compile you will not lose a point. If you believe that your code has incorrectly been marked as wrong you should inform a judge who will then have the question reviewed. You may make as many submissions for each question as you would like, although you will only receive credit for one correct answer per problem.

A real-life developer will grade problem 7. The rubric by which it will be graded is located at the back of this packet. Note that a non-working or partially working project can still receive points for design and organization. The final question will also be submitted through the website codeLM.com. Unlike the first six questions, you will upload the source code through the website as opposed to copying and pasting your code into a textbox. There are no restrictions to method and class names for the final problem.

Teams will be ranked by total points earned. If several teams earn the same amount of points, the teams will be ranked according to the following criteria:

1. Most correct answers.
   then
2. Least incorrect answers.
   then
3. Time last auto-graded solution was entered (earliest to latest).

Problem 1: Purchase Receipt (2 points)

New Wave Computers™ sells four products: **hard drives**, **software packages**, **wPhones** and **wPens**. Their retail division needs a program that will quickly calculate the total cost of a purchase from one of their stores. Their pricing scheme, as explained below, is very complicated.

Input
Four integers: **`drives`**, **`software`**, **`phones`**, and **`pens`**. These values represent the quantities of hard drives, software packages, wPhones, and wPens purchased.

Output:
The **total cost** of all of the products as a **`double`** or **`float`** value.

Pricing Scheme:
1. **Hard drives** cost $49.95 each. Customers receive 5% off the cost of the hard drives if more than 5 are purchased. They receive 10% off the cost of the hard drives if more than 10 are purchased. The 5% discount does not stack and the total discount at this level is 10%.

2. **Software packages** cost $25.00 each with $10 off the total software cost for each wPhone purchased. This discount can at most make the software cost $0; it cannot give the user money back.

3. **wPhones** cost $399.00 each.

4. The cost of a **wPen** is determined on a sliding scale which is described by the chart below:

Quantity Purchased	Price per Pen
1	$500
2	$450
3	$400
4	$350
5+	$300

5. A 6% **sales tax** should be added to all products.

Note
Your output will be checked to the nearest whole number. You do not need to worry about rounding or truncating your return value.

Sample Data

Input	Output	Explanation
1, 1, 1, 1	1021.78	Hard Drives: 49.95 Software: 15 wPhones: 399 wPens: 500 tax: 57.837 total 1021.787
8, 3, 4, 5	3721.25	Hard Drives: 379.62 Software: 35 wPhones: 1596 wPens: 1500 tax: 210.6372 total: 3721.2572
10, 3, 4, 8	4775.85	Hard Drives: 474.525 Software: 35 wPhones: 1596.0 wPens: 2400 tax: 270.3315 total 4775.8565
8, 2, 10, 10	7811.79	Hard Drives: 379.62 Software: 0 wPhones: 3990 wPens: 3000 tax: 442.1771 total: 7811.7972

Problem 2: New Wave Name (2 points)

New Wave Computers'™ marketing department would like to display the company's name on their wPhones and other small devices. You have been asked to write a program that will display arbitrary-length versions of their name for these small-screen devices.

Input
An integer, **n**, representing the number of characters of the name that are needed.

Output
n characters of the string "New Wave". If **n** is greater than the length of the string "New Wave", you must print the string, then a space, then the remaining characters. Note that the space counts as a character. Output should not end with a space. When the integer value, **n**, falls on a space, you should instead print a string of length **n-1**.

Sample Data

Input	Output	Notes
2	Ne	
9	New Wave	Note that this does not end with a space.
10	New Wave N	
25	New Wave New Wave New Wav	

Problem 3: Employee ID Verifier (3 points)

Due to a recent security breach, New Wave Computers™ has implemented a new staff identification system. They need help writing a program to validate IDs and then map valid IDs to a four-letter key.

Input
An 8-digit integer, **`id`**, representing an employee's ID.

Output
If the 8-digit integer ID does not match New Wave's prescribed security algorithm you should return the string "**invalid**". If the ID is valid then you should return the 8-digit ID converted to **a four-character string**.

Algorithm
The following conditions must be met for an ID to be valid:

1. The sum of all digits must be a multiple of 5.
2. All even-indexed numbers (second, fourth, etc.) must be odd.
3. The fourth digit must be within 2 of the second and sixth digits.

If the above conditions are not met, the result of the algorithm is the word "invalid". If the above conditions are met, the ID must be converted into a four-letter key. The four-letter key is the result of the following algorithm:

1. Split the 8-digit ID into four 2-digit numbers.
2. Convert each of the 2-digit numbers into the corresponding lowercase letter, making sure to restart the numbering if the number is above 26. (01 would convert to "a", 26 would convert to "z", and 27 would convert to "a").

Sample Data

Input	Output	Explanation
23456789	invalid	The sum of all digits is not a multiple of 5.
12759745	invalid	The second digit is not odd.
73294717	invalid	The fourth digit is not within 2 of the second digit.
25354565	yism	All criteria are met, so 1. 25 35 45 65 2. y i s m 3. yism
25432167	yquo	All criteria are met, so 1. 25 43 21 67 2. y q u o 3. yquo

Problem 4: Loading Symbol (3 points)

You are designing an hourglass loading symbol for New Wave Computers'™ new patent-pending word processor. This symbol will consist of asterisks. The loading icon needs to be resizable in order to work well on screens of different sizes.

Input
An integer, **n**, representing the number of rows the hourglass should have.

Output
If **n** is odd, the first row should be **n** asterisks long, with each succeeding row having 2 fewer asterisks with spaces to center it until there is only 1 asterisk, at which point the pattern is reversed until there are **n** asterisks again. There will only be one row that has a single asterisk.

If **n** is even, the first row should be **n-1** asterisks long, with each succeeding row having 2 fewer asterisks with spaces to center it until there is only 1 asterisk, at which point the pattern is reversed until there are **n-1** asterisks again. There will be exactly two rows that have a single asterisk.

Note
There should not be any spaces between asterisks. Your output for this problem will be printed to the console. Your method will not return anything; it is a void method.

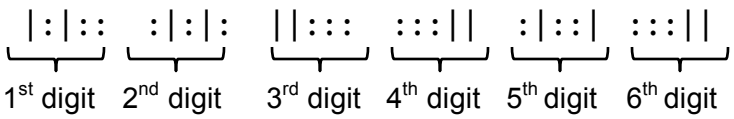
Sample Data

Input	Output
1	*
2	* *
3	*** * ***
4	*** * * ***
5	***** *** * *** *****

6	*****
	 ***
	  *
	  *
	 ***
	*****

Problem 5: Barcode (4 points)

New Wave Computers™ places a six-digit barcode on each of their products that looks like the following:



Each digit in the barcode is comprised of full bars '|' and half bars ':'. Each digit is separated by a single space.

To decode a digit, convert each full bar to a 1 and each half bar to a 0. Then, use the table below to turn the sequences of 0s and 1s into an integer.

	7	4	2	1	0
1	0	0	0	1	1
2	0	0	1	0	1
3	0	0	1	1	0
4	0	1	0	0	1
5	0	1	0	1	0
6	0	1	1	0	0
7	1	0	0	0	1
8	1	0	0	1	0
9	1	0	1	0	0
0	1	1	0	0	0

Note that they represent all combinations of two full and three half bars. The digit can be computed easily from the bar code using the column weights 7, 4, 2, 1, 0. For example, 01100 is

$$0 * 7 + 1 * 4 + 1 * 2 + 0 * 1 + 0 * 0 = 6$$

The only exception is 0, which would yield 11 according to the weight formula.

The first five encoded digits are followed by a check digit, which is computed as follows: Add up the first five digits, and choose the check digit to make this sum a multiple of 10. For example, the sum of the digits in the bar code 95014 is 19, so the check digit is 1, to make the sum equal to 20. Write a method that accepts a barcode as input and returns whether or not the barcode's check digit is valid.

Input
String barcode - A 39 character String of '|'s, ':'s and spaces.

Output
A boolean value which is true if the barcode's check digit is valid and false if the barcode's check digit is not valid.

Sample Data

Input	Output	Explanation
|:::| ::||: |::|: :::|| ||::: :|:|:	false	1. The barcode translates to 10001 00110 10010 00011 11000 01010 2. Which becomes 7 3 8 1 0 5 3. Check digit would have to be one to be correct, but it is five.
:::|| :|:|: |::|: :|::| ::|:| ::||:	false	1. 00011 01010 10010 01001 00101 00110 2. 1 5 8 4 2 3 3. Check digit would have to be seven to be correct, but it is three.
||::: |:::| :||:: :|:|: :::|| :::||	true	1. 11000 10001 01100 01010 00011 2. 0 7 6 5 1 1 3. Check digit is correct, so return true.

Problem 6: Weekly Report (4 points)

Due to recent inconsistencies observed by the SEC, New Wave Computers™ must release weekly earnings reports. Unfortunately these reports often contain several mistakes. The SEC also requests that all monetary values be converted to euros so they can better analyze New Wave's oversea transactions.

Input
A string, **`report`**, containing the full report.

Output
The same report, but with the following modifications made:

- The first letter of every sentence must be a capital letter, including the first sentence in the report. A sentence can end with either a period, a question mark, or an exclamation point. There will always be exactly one space between a punctuation mark and the first letter of the next sentence.
- In all instances of "mr", "ms", or "mrs", the m should be capitalized. All instances of these titles will be followed with a period and a space.
- All dollar values should be converted to euro values. A dollar value is a dollar sign followed by one or more integer digits with no spaces. The dollar sign should be replaced with a euro sign (we'll just use a capital E), and the value should be converted based on the equation *euro* = .75 * *dollar*. All numeric values greater than 999 will be written without commas and all dollar values will be whole numbers.

Sample Data

Input	Output
here at New Wave, we are proud to launch our brand new wPhone 2, starting at just $500. our CEO mr. swope is very proud of our teams hard work.	Here at New Wave, we are proud to launch our brand new wPhone 2, starting at just E375. Our CEO Mr. Swope is very proud of our teams hard work.
in our last quarter, we made a profit of $375 per wPhone and $10 per software package. our stock price also grew 3.1%.	In our last quarter, we made a profit of E281.25 per wPhone and E7.5 per software package. Our stock price also grew 3.1%.
employee complaint #687517: mr. John has been acting up again. i cannot believe he makes $10000 a year. fire him. now!	Employee complaint #687517: Mr. John has been acting up again. I cannot believe he makes E7500 a year. Fire him. Now!

Problem 7: Diecisiete (8 points)

New Wave Computers'™ Gaming Division's newest release will be the game of **Diecisiete**. This game is played using a modified Uno© deck. The object of the game of Diecisiete is to beat the computer in one of the following ways: reach a final score higher than the computer without exceeding 17 or be dealt 5 cards without exceeding a value of 17.

Rules
At the start of the game the player and computer are each dealt two cards. The computer and player take turns throughout the game either drawing another card or staying. Once the computer or player has decided to 'stay' they can no longer 'draw' additional cards. Play will continue until the other contestant either 'stays', is dealt five cards or exceeds a sum of 17. The player will be given the option to either stay or draw throughout the game. The computer should continue to draw until it's hand has a total greater than 13 or it can see that the player has exceeded 17. The computer cannot 'see' the first card in the player's hand so the summation that the computer uses to to determine if it should draw or stay should not take this first card into account.

The modified Uno deck contains 80 cards. There are four suits: red, green, yellow and blue. Each color consists of one **0** card, two **1**s, two **2**s, two **3**s, two **4**s, two **5**s, two **6**s, two **7**s, two **8**s and two **9**s. The player's and dealer's score is calculated by summing the values of each card in their hands. When calculating this sum each rank card has a value equal to its rank.

During game play you cannot see the dealer's first card but will be able to see any card after this. Cards should be randomly dealt. You do not need to keep track of which cards have already been dealt, but should instead generate a random number between 0 and 3 for the card's color (0 – red, 1 - green, 2 - blue, 3 – yellow) and a random number between 0 and 9 for it's value. Zero through nine represent a card with ranks from zero to nine.

An asterisk should represent the dealer's first card. Wilds should be represented with just a 'W'. All other cards should be represented by a combination of either the letter 'R', 'B', 'G' or 'Y' and it's 1 through 9 value.

Generating a Random Number – The following code will generate a random number between 1 and `maxNumber`.

Java
```
int index = (int) (Math.random()*maxNumber-1);
```

Python
```
import random
index = random.randint(1, maxNumber)
```

C++
```
/*must import time.h */
/* initialize random seed: */
srand (time(NULL));
```

```
/* generate secret number between 1 and maxNumber: */
index = rand() % maxNumber + 1;
```

Note

This problem will be graded by a real-life developer. The rubric by which it will be graded is at the back of this packet. Note that a non-working or partially-working project can still receive points for design and organization.

Sample game 1:

```
Welcome to Diecisiete.

Computer’s hand: * B3
Your hand: G6 G9

The computer has chosen to draw another card.

Computer’s hand: * B3 B1
Your hand: G6 G9

Enter a 1 to draw or a 2 to stay.
1

Computer’s hand: * B3 B1
Your hand: G6 G9 Y2

The computer has chosen to draw another card.

Computer’s hand: * B3 B1 G7
Your hand: G6 G9 Y2

Enter a 1 to draw or a 2 to stay.
2

Computer’s hand: Y5 B3 B1 G7
Your hand: G6 G9 Y2

The computer’s total is 16 Your total is 17

You have more points than the computer and didn't go over 17.  You win!
```

Sample game 2:

```
Welcome to Diecisiete.

Computer’s hand: * G6
Your hand: B6 R7

The computer has chosen to draw another card.

Computer’s hand: * G6 G8
Your hand: B6 R7

Enter a 1 to draw or a 2 to stay.
2

The computer has chosen to stay

Computer’s hand: Y0 G6 G8
Your hand: B6 R7

The computer’s total is 14 Your total is 13

The computer wins.
```

Sample game 3:

```
Welcome to Diecisiete.

Computer’s hand: * Y8
Your hand: B1 B4

The computer has chosen to draw another card.

Computer’s hand: * Y8 B5
Your hand: B1 B4

Enter a 1 to draw or a 2 to stay.
1

Computer’s hand: * Y8 B5
Your hand: B1 B4 Y0

The computer has chosen to stay

Computer’s hand: * Y8 B5
```

```
Your hand: B1 B4 Y0

Enter a 1 to draw or a 2 to stay.
1

Computer’s hand: * Y8 B5
Your hand: B1 B4 Y0 G3

The computer has chosen to stay

Computer’s hand: * Y8 B5
Your hand: B1 B4 Y0 G3

Enter a 1 to draw or a 2 to stay.
1

Computer’s hand: * Y8 B5
Your hand: B1 B4 Y0 G3 B0

The computer has chosen to stay

Computer’s hand: * Y8 B5
Your hand: B1 B4 Y0 G3 B0

Enter a 1 to draw or a 2 to stay.
2

Computer’s hand: Y4 Y8 B5
Your hand: B1 B4 Y0 G3 B0

The computer’s total is 17 Your total is 8

You got 5 cards and didn't go over 17.  You win!
```

Diecisiete Rubric:

	0 points	1 point	2 points
Function	A program solution is submitted but fails to compile. -- or - The submitted program compiles successfully. * The program does simulate playing the game of Diecisiete.	The submitted program compiles successfully. * The submitted program includes run-time and/or logic errors that result in incorrect output. * Implementation is incomplete.	The submitted program compiles successfully. * The submitted program is free of run-time and logic errors. * Your code fully implements the game of Diecisiete.
Code Readability	Code contains no documentation. * Code is unformatted and is difficult to read. * variables are ambiguous (i.e. x) and do not indicate the purpose of the variable.	The submitted solution is inconsistently documented. * Code is inconsistently formatted and can be difficult to read. * Numerous variables are ambiguous (i.e. x) and do not indicate the purpose of the variable.	The submitted solution is well documented. * Code is properly formatted (i.e. indentation within brackets and appropriate spacing) and is easy to read. * All variables are self-documented (i.e. named in a way that the name indicated the purpose of the variable).
Design	Code shows little to no thought about design. * Structures and data types are poorly chose..	Code shows some thought about design. * Appropriate structures are often used but not consistently throughout the program.	The program effectively chooses and implements concepts that would best model and solve the problem. * Appropriate data types are chosen for all variables. * Code uses the most appropriate structures (i.e. if, else if, else, methods and loops.)
Interface	The user interface demonstrates appropriate spacing and descriptive instructions. * Cards are not displayed correctly. * Game flow is incorrect..	The user interface includes minor spacing problems which result in inconsistent or confusing input/output. * Cards are displayed, but there may be some inconsistencies or errors. * Game flow is inconsistent.	The user interface demonstrates appropriate spacing and descriptive instructions. * Cards are properly displayed. * Game flow is correct.