(Lab 1) Problem Set 1:
Data in → Processing → Data out

P1 Solutions limited in scope to:		
● input/scanner ● output/print ● storage/variable	● arithmetic ● modulus ● logical	● relational ● equality ● casting

Submission Rules:

1. Submissions must be zipped into a **handin.zip** file. Each problem must be implemented in its own class file. Use the name of the problem as the class name.

2. You must use standard input and standard output for ALL your problems. It means that the input should be entered from the keyboard while the output will be displayed on the screen.

3. Your source code files should include a comment at the beginning including your name and that problem number/name.

4. The output of your solutions must be formatted exactly as the sample output to receive full credit for that submission.

5. Compile & test your solutions before submitting.

6. Each problem is worth up to 10 points total. The breakdown is as follows: 2 points for compiling, 3 points for correct output with sample inputs, 5 points for additional inputs.

7. This lab is worth a max total of: 40 points. You can complete as many problems as you like, but cannot receive more than 40 points towards the lab grade. All points in excess of that are for bragging rights. (Check the scoreboard to see how you did!)

8. Submission:

- You have unlimited submission attempts until the deadline passes

- You'll receive your lab grade immediately after submitting

- **IMPORTANT:** if your grade is lower than 70% when the deadline passes, then you must attend a recitation session & get TA signoff to receive full credit for that lab challenge.

Problem 1: Draughting Dollars (10 points)

You're the new manager of a craft beer pub. On your first day, the ATF demands an audit of your beer inventory in dollar amounts. The problem is that your on-tap kegs are only partially filled. You must determine the current retail worth of a tapped keg, based on the percentage of beer remaining, and the price of the beer per pour.

Facts:

- A full-sized keg is 15.5 gallons.
- There are 128 ounces in 1 gallon.
- Beer is served in a pint glass
- There are 16 ounces in 1 pint.

Input

Your solution must take in two inputs. The first input represents the percentage of beer remaining in the keg represented as a decimal ratio, with valid ranges from 0.0 (empty) to 1.0 (full). The second input represents the price of the beer per serving, as dollars and cents.

Output

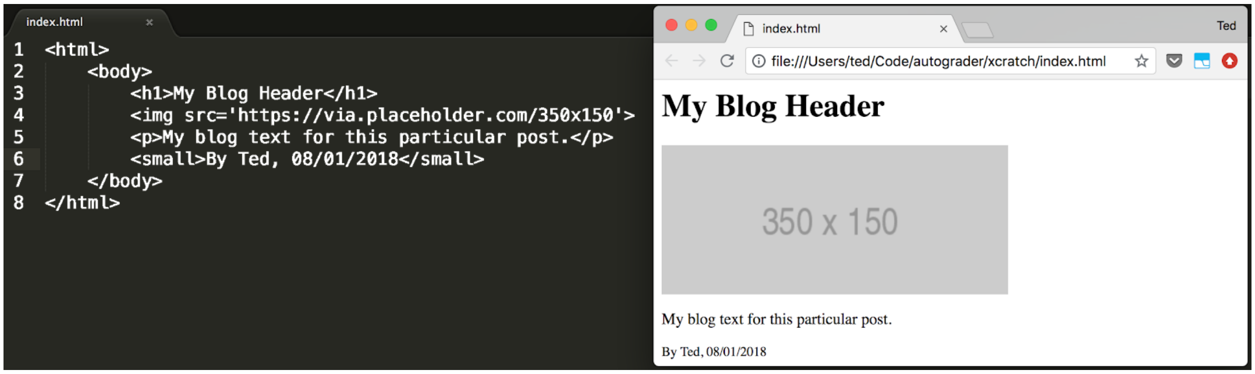
The output should display the following text: "There is $<dollars> of beer left in keg" where <dollars> is the numerical value of how much is left.

Sample Input	**Sample output**

0.5 4
1.0 6
0 8
	There is $248.00 of beer left in keg
	There is $744.00 of beer left in keg
	There is $0.00 of beer left in keg

Problem 2: Blogging in HTML (10 points)

Internet browsers, such as Chrome or Firefox, are applications designed to open and view HTML files. HTML is the basic formatting rules used for all web pages. In an HTML file, tags surround text elements to tell the browser how that text should be displayed. Suppose that you're a blogger and want to convert your plain text into an HTML blog post which consists of the following: topic header, image, text paragraph, & accreditation. See below for an illustration:



```
<html>
    <body>
        <h1>My Blog Header</h1>
        <img src='https://via.placeholder.com/350x150'>
        <p>My blog text for this particular post.</p>
        <small>By Ted, 08/01/2018</small>
    </body>
</html>
```

My Blog Header

My blog text for this particular post.

By Ted, 08/01/2018

Facts:

● HTML tags	*<html></html>*	All html data goes between html tags
● Body tags	*<body></body>*	All text/images goes between body tags
● Heading tags	*<h1></h1>*	Header text goes between heading tags
● Image tag	*<img src="path/to/file" />*	Image file path goes within an image tag
● Paragraph tags	*<p></p>*	Basic text goes between paragraph tags
● Small tags	*<small></small>*	Small text goes between small tags

Input

Your solution must take in five text inputs. Each input is on its own line and may contain multiple words. The first input represents the blog's header. The second input is the image's file path. The third input is the post's text. The fourth input is the blog's author. The fifth input is the post's date formatted as *mm/dd/yyyy*.

Output

The output should produce the appropriate HTML text for the blog post. See the sample output below as a guide. There should be no spaces between html elements.

Sample Input	**Sample output**
Header image.png text description me 08/01/2018	<html><body><h1>Header</h1><img src='image.png' /><p>text description</p><small>By me, 08/01/2018</small></body></html>

Problem 3: Papers Please (10 points)

You're a ticket agent for a commercial airline and responsible for checking the identification for each passenger before issuing their boarding pass. Due to federal regulations, there is a total of three different forms of ID that can potentially be used. You must determine whether the passenger has the sufficient identification to board the plane or not.

Facts:

- A passport is enough for a boarding pass
- Without a passport, passengers must have two other forms of ID:
    - driver's license
    - birth certificate
- If the above two conditions are not met, then they are denied.

Input

Your solution must take in three boolean inputs. The first input represents whether they have a passport or not. The second input represents whether they have a driver's license or not. The third input represents whether they have a birth certificate or not.

Output

The output should display as a boolean result whether they can board the plane or not.

Sample Input	**Sample output**
true false false false true true false false false	true true false

Problem 4: Bouncer Bot (10 points)

You're hired by the popular nightclub, 'Club B-day' to automate the job of their bouncers.
Club B-day has a novel criteria for entering, where it must be that person's birthday. Additionally state law stipulates that a patron must also be aged 21 or over to enter the premises since alcohol is served. Your robot must determine whether a person can enter the club given the following conditions:

- Is today their birthday?
- Are they 21 years or older?

Input

Your solution must take in a total of six numerical inputs with each input separated by a space. The first three inputs represent the current month, current day, current year as integers. The other three inputs represents the birth month, birth day, birth year as integers.

Output

The output should display as a boolean result whether the patron can enter or not.

Sample Input	**Sample output**

1 31 2018 1 31 1990
1 31 2018 1 30 1990
1 31 2020 1 31 2010
	true
	false
	false

Problem 5: Student Grader (10 points)

You're a student enrolled in the introductory computer science course at UNO and you just can't wait to see what your grade is so you decide to calculate it on your own. The problem is that your final grade is determined as a weighted average between three different grade types: Test average, Lab average, Homework average.

Facts:

- Test average is worth 40% of final grade
- Lab average is worth 10% of final grade
- Homework average is worth 50% of final grade

Input

Your solution must take in three numerical inputs (floating point) with a range from 0.0 to any positive number. The first input represents the tests average. The second input represents the homework average. The third input represents the lab average.

Output

The output should display the student's final average as a numerical score (floating point).

Sample Input	**Sample output**

90 80 70
75 100 100
0 0 0
	83.0
	90.0
	0.0

Problem 6: ASCII encodings (10 points)

People read/write text using alphanumeric characters *(i.e. 'a', 'b', 'c')* whereas computers are limited to processing purely numerical data. Therefore, in order for machines to process text, that text must first be encoded as numbers. ASCII is the most common mapping between characters and numbers. Your task is to take the raw numerical data used by a computer, and convert it into an english message that a person may easily read.

Facts:

- the char datatype maps characters as ascii codes
- ascii codes are integer numbers
- Java may easily convert between char codes and integers

Input

Your solution must take in six integer inputs. Each input represents an alphanumeric character.

Output

Your solution must display the six-lettered english message as a single string with no spaces between the characters.

Sample Input	**Sample output**
72 101 108 108 111 33	Hello!
87 111 114 108 100 46	World.
67 83 64 85 78 79	CS@UNO
104 97 99 107 101 114	hacker

Problem 7: How odd! (10 points)

Integer numbers are either even or odd. One of the first mathematical skills that's taught to children is to identify whether a number is even or odd. Write a program that can determine whether an integer is odd or not.

Facts:

- Even numbers may be mathematically defined as *2n*, where *n* is any integer number
- Odd numbers may be mathematically defined as *2n+1*, where *n* is any integer number

Input

Your solution must take in a single integer of any possible value.

Output

Your solution should output as a boolean result: `true` if the number is odd or `false` if its even.

Sample Input	**Sample output**
2 3 100000001	false true true

Problem 8: Around the Clock (10 points)

You've been hired to work with air traffic control to determine the expected arrival time of an incoming aircraft. You must determine the arrival time using only the aircraft's departure time and the travel time.

Facts:

- All times are given in hour blocks, no minutes, no seconds
- Don't worry about time zones
- Air traffic control uses a 12 hour cycle to determine time.
- You may treat the result of 0 o'clock as 12 o'clock

Input

You will be given two integer inputs. The first input is the departure time in an am/pm cycle (1 to 12). The second input is the travel time in hours, (any positive number).

Output

Your solution must print the integer result that represents the expected arrival time in an am/pm cycle. Therefore your output should be limited between 0-11

Sample Input	Sample output

11 4
4 8
7 36
	3
	0
	7

Problem 9: Duck Duck Goose (10 points)

Duck, duck, goose is a child's game where kids sit in a circle, facing inward, while another player, walks around calling each a "duck" until finally calling one a "goose". Given the number of kids seated in a circle, and the number of ducks called out, determine which child is the goose.

Facts:

- The player circles the group in clockwise rotation
- Label the children numerically.
    - The first duck called is labeled: child 0
    - To the left of child 0 is child *n-1*, where *n* is the number of children

Input

You are given two positive integer inputs. The first represents the number of kids in the circle. The second represents the number of ducks that are called before goose.

Output

Print as an integer which child was selected as goose.

Sample Input	Sample output

7 15
4 4
12 6
	1
	0
	6