CSE 101: Introduction to Computational and Algorithmic Thinking

Stony Brook University

Homework Assignment #1

Spring 2018

Assignment Due: February 5, 2018 by 11:59 pm

Assignment Objectives

By the end of this homework assignment you should be able to writing short Python functions that feature variables and mathematical expressions.

Getting Started

Visit Piazza and download the "bare bones" file `homework1.py` onto your computer, as well as `homework1_driver.py`. Open `homework1.py` in PyCharm and fill in the following information at the top:

1. your first and last name as they appear in Blackboard

2. your Net ID (e.g., jsmith)

3. your Stony Brook ID # (e.g., 111999999)

4. the course number (CSE 101)

5. the assignment name and number (Homework #1)

Do not, under any circumstances, change the names of the functions or their argument lists. The automated CodeLoad testing system will be looking for exactly those functions provided in `homework1.py`. You will be able to test your work by uploading your file to CodeLoad.

Submit your final `homework1.py` file to Blackboard by the due date and time. Late work will not be graded.

Code that crashes and cannot be graded will earn no credit. It is your responsibility to test your code by running it through CodeLoad and by creating your own test cases.

Part I: Compute the Value of $\dfrac{x}{y - 3x} + \dfrac{x - 1}{\frac{y}{x+1}}$ (20 points)

Write a function `compute1()` that takes the following arguments, in this order:

1. `x`: the first value you need for the formula

2. `y`: the second value you need for the formula

The function computes and returns the value of the formula given below (using floating-point division):

$$\frac{x}{y - 3x} + \frac{x - 1}{\frac{y}{x+1}}$$

Note: You don't need to worry about argument values that would cause division by zero errors.

Examples:

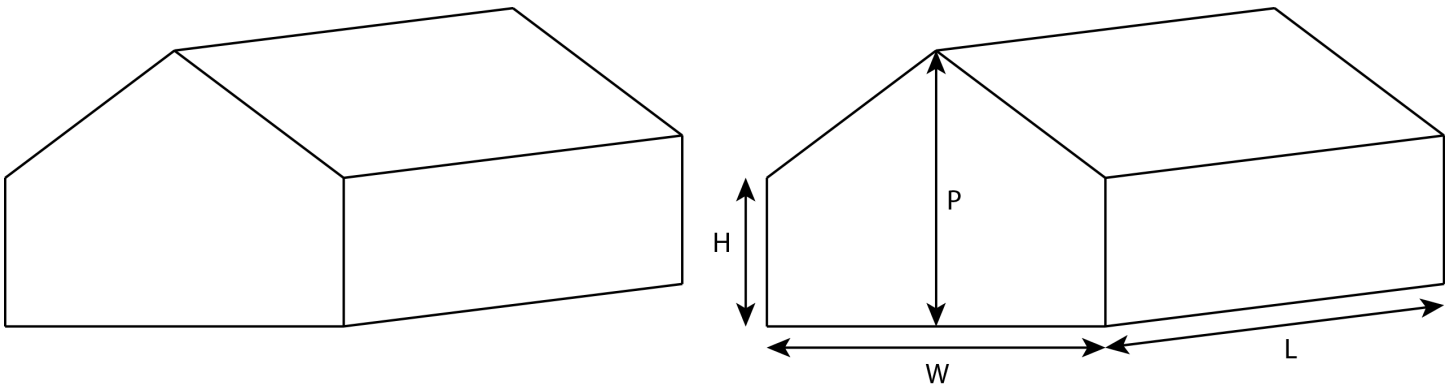| Function Call	Return Value
`compute1(6, -3)`	`-11.952380952380954`
`compute1(93.33, 15)`	`580.2803913634477`
`compute1(4, 5.3)`	`2.2331737538721486`

Remember: CodeLoad has additional tests for you to try! Upload your code there and see how your code matches up against harder tests.

Part II: Space in the Farmhouse (20 points)

Write a function `farmhouse()` that takes the following arguments, in this order:

1. `W`: the width of a farmhouse, as illustrated in in the figure below

2. `L`: the length of the farmhouse

3. `H`: the height of the lowest point of the roof

4. `P`: the height of the peak of the roof

The function computes and returns the volume of space occupied by a farmhouse of the provided dimensions. You may assume $P > H$ and that all five values are positive. Do not assume that $H = P/2$. Use floating-point division only.

Examples:

Function Call	Return Value
`farmhouse(6.25, 17, 8, 9.7)`	`940.3125`
`farmhouse(19, 28, 10, 12)`	`5852.0`
`farmhouse(10, 40.5, 11, 16)`	`5467.5`

Remember: CodeLoad has additional tests for you to try! Upload your code there and see how your code matches up against harder tests.

Part III: Car Fuel Consumption (20 points)

Write a function `fuel()` that takes the following arguments, in this order:

1. `minutes`: the number of minutes a car is driven on a long track

2. `mph`: the constant speed of the car, given in miles per hour

3. `mpg`: the fuel efficiency of the car, given in miles per gallon

The function computes and returns the number of gallons of gasoline consumed by a car driven down a long track over the given time period. Use floating-point division only.

Examples:

Function Call	Return Value
`fuel(75, 35.2, 20.6)`	2.1359223300970873
`fuel(60, 60, 30)`	2.0
`fuel(164.3, 45.2, 19.41)`	6.376747381075049

Remember: CodeLoad has additional tests for you to try! Upload your code there and see how your code matches up against harder tests.

How to Submit Your Work for Grading

To submit your `.py` file for grading:

1. Login to Blackboard and locate the course account for CSE 101.

2. Click on "Assignments" in the left-hand menu and find the link for this assignment.

3. Click on the link for this assignment.

4. Click the "Browse My Computer" button and locate the `.py` file you wish to submit. Submit only that one `.py` file.

5. Click the "Submit" button to submit your work for grading.

Oops, I messed up and I need to resubmit a file!

No worries! Just follow the above directions again. We will grade only your last submission.

Note: Files uploaded to CodeLoad will **not** be graded! You must submit your work to Blackboard for grading.