Assignment #1A (due 9/13 at 11:59 PM)

* **Due** Sep 13 by 11:59pm  **Points** 20

**Submitting** a file upload

**Available** Sep 8 at 12am - Nov 19 at 11:59pm 2 months

Code a Python program the does the following:

The program accepts two inputs from the keyboard, converts them to floats, and stores them in separate variables called 'length' and 'width'.  Those variables represent the dimensions of a rectangle.

Use those values to calculate and display the perimeter and area of the rectangle.

Include a comment at the top of your code that contains your full name and the date on which you coded your program.

Example of how the program might run:

**Enter the length of a rectangle: 5**  
**Enter the width of a rectangle: 3**  
**The area of the rectangle is: 15.0**  
**The perimeter of the rectangle is: 16.0**

Assignment #1B (due 9/13 at 11:59 PM)

* **Due** Sep 13 by 11:59pm  **Points** 20

**Submitting** a file upload

**Available** Sep 8 at 12am - Nov 19 at 11:59pm 2 months

Code a Python program the does the following:

Get user input from the keyboard and convert it into a float.  That number represents a temperature in Fahrenheit.  Then convert that temperature to Celsius and display it.

Here's the conversion formula:  Celsius = (Fahrenheit − 32) \* 5/9

As a test, if you enter a value or 212 degrees Fahrenheit, then the converted value should be 100 degrees Celsius.

If you run the program, it might look like this:

**Enter a temperature in Fahrenheit:  212**  
**The temperature in Celsius is: 100.0**

Note the 100.0 -- that's because you are displaying a float value.

Assignment #2A (due 9/20 at 11:59 PM)

* **Due** Sep 20 by 11:59pm

**Points** 20

**Submitting** a file upload  **Available** Sep 14 at 12am - Nov 19 at 11:59pm 2 months

Create a program where the user is asked if they wish to add, subtract, multiply, or divide.  Then get two float values from the user.  Then perform the appropriate mathematical operation and display the results.

Possible run of the program:

**Welcome to a simple calculating program!**

**Enter a 1) to add, 2) to subtract, 3) to multiply, or 4) to divide:  1**

**Enter your first value:  2.35**

**Enter your second value:  3.31**

**Results: 5.66**

Assignment #2B (due 9/20 at 11:59 PM)

* **Due** Sep 20 by 11:59pm  **Points** 20  **Submitting** a file upload

**Available** Sep 14 at 12am - Nov 19 at 11:59pm 2 months

Write a program for the following problem.

This program will calculate the individual pay and total pay for the employees of a company.  The program will loop until the user tells it to stop.  Each run of the loop, the user will enter an employee's pay-rate and hours worked that week.  At this time, don't worry about validating those values, but do give the user clear instructions that they should enter appropriate values (see the example).

Each employee's individual pay should be calculated and displayed (i.e. pay-rate multiplied by hours).  Remember to calculate for overtime pay (see notes).

Accumulate the individual pay values for the employees.  When the loop ends, display the total pay for all employees.

**An example of how your program might run:**

**Enter employee's pay-rate (more than $9.25): 10**

**Enter hours worked this week (zero or a positive number): 10**

**Employee's weekly pay: 100**

**Do you wish to continue? (yes/no): yes**

**Enter employee's pay-rate (more than $9.25): 10**

**Enter hours worked this week (zero or a positive number): 45**

**Employee's weekly pay: 475**

**Do you wish to continue? (yes/no): no**

**Total pay for all employees is: $575**

# What's Overtime Pay?

Assume overtime pay works like this.

If an employee works 40 or less hours per week, their weekly pay is the number of work-hours time their pay-rate.

Example:   An employee worked 20 hours in a work-week for $10.00 an hour.  Their final weekly pay is $200.00.

However, overtime pay is more complex.  If an employee works more than 40 hours in a week, they get their base pay-rate for the first 40 hours.  After 40 hours, the get one and a half times their pay-rate.

Example:  An employee works 45 hours in a week at $10.00 an hour.  Their final weekly pay is $475.00 (40 hours at $10 dollars per hour and 5 hours at $15 per hour).

# Assignment #2B wants accumulation. How do you do that?

This is a commented Python program that shows how to do accumulation.  You might want to copy and run it.

**## Accumulating a value can be tricky at first.  Here's an example**

**## Here's the variable that we will accumulate into**  
**total = 0**

**## This loop will run 3 times - from value 0 to value 2**  
**value = 0**  
**while value < 3:**  
**temp = int(input("Enter an integer what will be accumulated: "))**  
**total = total + temp ## bad algebra, but good computer programming**  
**value = value + 1**

**print("The accumulated total entered is: ", total)**

**## The loop features two statements that often bother people.**  
**##      total = total + temp**  
**##      value = value + 1**  
**## Both are examples of the 'replacement algorithm'.**  
**## They are bad algebra -- but good programming.**  
**## What we're saying is:**  
**##    "see this variable? (total or value in the above cases)"**  
**##    "Modify what's in that variable and assign the result back the variable:"**  
**##    So if total is 0 and temp is 12, then after we accumulate, the total will contain 12.**  
**##    if total is 12 and temp is 3, then after we accumulate, the total will contain 15.**

Assignment #3A (due 9/27 at 11:59 PM)

Re-submit Assignment

* **Due** Sep 27 by 11:59pm

**Points** 20

**Submitting** a file upload

**Available** Sep 21 at 12am - Nov 19 at 11:59pm 2 months

**In terms of being forced to make students redo an assignment, this is actually one of the most difficult assignments I give.  Note that some of these functions will ask you to print and others to return.   Pay close attention to slides 9 and 10 in the lecture for this week and make sure you understand the difference.**

Create a function called 'append' that takes two string parameters, concatenates them together, and then prints the concatenated string.  Note that this is a void function -- it does not return anything

Create a function called 'positiveNumber'.  It has a single parameter that should be a number.  If the parameter value is less than zero, return a zero.  Otherwise, return the original value.

Create a function called 'evenOdd'.  It has a single parameter that should be a number.  If the parameter value is even, return a true value.  Otherwise, return a false value (hint: the modulus operator -- the '%' -- could be useful).

Create a function called 'inCenturyRange'.  It has a single parameter that should be a number.  The function returns a true of the value between 0 and 100 (inclusive).  It returns false otherwise.

Create a function called 'repeater' that has two parameters -- a string and number.  Give the two parameters default values of your choice  The function concatenates the string a number of times equal to the number value.  It then prints the concatenated string.  The function doesn't return anything.

Remember to include code in your file that tests all of the above functions.

Note:  at this point, don’t worry about passing the wrong data-types into your functions – which will quite likely cause your code to fail.  However, if you wish to work ahead, consult the chapter in your book about Exception Handling.

Assignment #3B (due 9/27 at 11:59 PM)

* **Due** Sep 27 by 11:59pm

**Points** 20

**Submitting** a file upload

**Available** Sep 21 at 12am - Nov 19 at 11:59pm 2 months

Rewrite Lab 2A (from the previous week) so that it incorporates the following: there should be add, subtract, multiply, and divide functions.  Each function takes two arguments and returns the appropriate value based on those arguments.

Assignment #4A (due 10/04 at 11:59 PM)

* **Due** Oct 4 by 11:59pm

**Points** 20

**Submitting** a file upload

**Available** Sep 28 at 12am - Nov 19 at 11:59pm about 2 months

Create a program that does the following:

The user enters an integer value.  Then the program will create a list object that has the size of that integer value.

After that, code a loop that loops through the list, assigning an integer value from the keyboard to each element of the list.  If the user tries to enter zero or less, assign a one instead.  If the user tries to assign a number greater than 10, assign a 10 instead.

After the list is loaded, loop through it again, printing a number of asterisks equal to each element's number value.

Here's a possible output for the above program:

**Hello, please enter an integer value the size of your list:  3**  
**Please enter an integer value between 1 and 10 (inclusive):  2**  
**Please enter an integer value between 1 and 10 (inclusive):  12**  
**Please enter an integer value between 1 and 10 (inclusive):  7**

**Printing your entries as a chart:**  
**\*\***  
**\*\*\*\*\*\*\*\*\*\***  
**\*\*\*\*\*\*\***  
   
(Note that the second bar of the chart consists of 10 asterisks -- not 12).

Assignment #4B (due 10/04 at 11:59 PM)

* **Due** Oct 4 by 11:59pm
* **Points** 20
* **Submitting** a file upload  **Available** until Nov 19 at 11:59pm

In this exercise, we will create two parallel arrays, populate them (using a loop) and then display the results.

Create two 10 element lists.  The first list is intended to contain customer names.  The other list will contain the amount each customer purchased during a set period of time.

You will populate both lists with a single loop.  In each iteration of the loop, the user is asked for a name and a dollar amount.  Those values are assigned to incrementing elements of the list.

So in the first run of the loop, you will assign a name to the first element of one list, and a dollar amount to the first element of the second list.  In the second run of the loop, you assign a name to the second element of one list, and a dollar amount to the second element of the second list.

When you are done, print a report that contains the following -- each customer's name and how much they spent.  The total of all cash spent, the average of all cash spent, the largest amount spent, and the least amount spent.

Assignment #5A (Due 10/11 at 1:59 PM))

* **Due** Nov 19 by 11:59pm  **Points** 20

**Submitting** a file upload

**Available** Oct 5 at 12am - Nov 19 at 11:59pm about 2 months

Create a writeToTextFile function.  It takes two parameters that consist of a file path and a string that you plan to append to the text file described in the file path parameter.  The function returns nothing.

Create a readFromTextFile function.  It takes a single parameter that is a file path.  The function then returns a string that is the contents of the indicated text file.

Create a readListFromTextFile function.  It takes a single parameter that is a file path.  The function then returns a list that contains the contents of the indicated text file.  Each element of the list contains an individual line of text from the text file.

Create an application that can write text to, or read text from, a text file.  Use the three functions above and make sure you display what the read functions return.  The application should have a menu and run continuously until the user tells it to stop.