Important

1. Due Date: **Thursday, January 19th**
2. This homework is graded out of 100 points.
3. This is an Individual Assignment. You may collaborate with other students in this class. Collaboration means talking through problems, assisting with debugging, explaining a concept, etc. Students may only collaborate with fellow students currently taking CS 1301, the TA's and the lecturer. You should not exchange code or write code for others. For individual assignments, each student must turn in a unique program. Your submission must not be substantially similar to another student's submission. Collaboration at a reasonable level will not result in substantially similar code.
4. For Help:
   - TA Helpdesk (Schedule posted on class website.)
   - Email TA's or use Piazza Forums Notes:
   - How to Think Like a Computer Scientists [http://openbookproject.net/thinkcs/python/english3e/]
   - CS 1301 Python Debugging Guide [http://www.cc.gatech.edu/classes/AY2016/cs1301_spring/CS-1301-Debugging-Guide/index.html]
5. Don’t forget to include the required collaboration statement (outlined on the syllabus). Failing to include the Collaboration Statement will result in no credit.
6. Do not wait until the last minute to do this assignment in case you run into problems.
7. **Read the entire specifications document before starting this assignment.**

Introduction

The goal of this homework is for you to allow your TAs to get to know you as well as allow you to practice and understand how to write functions and evaluate expressions. Part 1 is simply uploading a photo of your face given the specifications listed. Part 2 will consist of 7 functions for you to implement. **You have been given HW1.py** to fill out with instructions in the docstrings. However, below you will find **more detailed information** to complete your assignment. Read it thoroughly before you begin. You have until **Thursday, January 19** to complete this assignment.

String Formatting

A concept that will be very helpful for this homework is **string formatting**. The purpose of this is to allow you to manipulate statements using variables so that values can change within depending on the different outputs of the function. In order to do this, let’s use an example where I ask the user for their name, and then print it out. Try this examples out:

```
name = input(“What is your name?”)
print(“Your name is {}”.format(name))

num = 3.141592
print("pi is {:.2f}".format(num))
```

What is happening is that anywhere in a string, I can put `{}` to indicate a placeholder for a variable. After the end quotation marks, I would write `.format()` to the end, and inside the parenthesis will be the variables that you want to include, in the order that `{}` are placed in the sentence. Read this for more info: [https://pyformat.info/]

Part 1: Mugshot

Find a picture of yourself. Crop the picture to include just your face. (Your face must be at least 1/3 of the image.) Resize the picture to be exactly 300 pixels by 300 pixels in size. Save the picture as a JPG, using your first and last name as the file name, (e.g. Jay_Summet.jpg). Submit this picture to t-square. You may use an image editing tool you are familiar with to manipulate your mugshot picture. If you do not have a preference, we recommend using the GNU Image Manipulation Program (GIMP): http://www.gimp.org/downloads/

Part 2: Functions

Function name (1): **circleArea**
Parameters: **N/A**
Return value: **N/A**
Description:
Write a Python program which accepts the radius of a circle from the user and compute the area.

1. Ask the user to input the radius of the circle.
2. In geometry, the area enclosed by a circle of radius r is πr^2. Calculate the area, round it to two three points and print a descriptive prompt. (Ex. `The area of the circle with radius 1.1 is 3.801`)

Hints:

- You can round a number by using the round(number, number of decimal places) function or by using string formatting. For example, you can round 56.789 to 56.79 by using `round(56.789,2)`.
- Importing the `math` module to get the value of π might be helpful.

Function name (2): **howManyLegs**
Parameters: **N/A**
Return value: **N/A**
Description:
Write a function that takes the number of chickens, tarantulas, and horses a person owns and calculates the number of total legs among all the animals.

1. Get the number of chickens the user owns; multiply the number by 2.
2. Get the number of tarantulas the user owns; multiply this number by 8.
3. Get the number of horses the user owns; multiply this number by 4.
4. Add all these numbers together and print the total in a descriptive format. (Ex. `4 chickens, 2 tarantulas, and 6 horses have 48 legs.`)

Function name (3): **calcBMI**
Parameters: **N/A**
Return value: **N/A**
Description:
Write a function that takes in a person's height and weight and calculates a person's body mass index (BMI).

1. Get the person's weight in pounds from the user; make sure to use a descriptive prompt so the user knows what units to use.
2. Get the person's height in inches from the user; again, make sure to use a descriptive prompt so the user knows what units to use.
3. Calculate the person's BMI as following:

a. Multiply the person's weight by .45 to convert it to kilograms.
b. Multiply the person's height by .025 to convert it to centimeters, then square the result.
c. Divide the result from step *a* by the result from step *b*.
d. Print out the calculated BMI, rounded to one decimal place, in a descriptive format. (Ex. `A person who weighs 140 pounds and is 66 inches tall has a BMI of 23.1.`)

Function name (4): **publix**
Parameters: **N/A**
Return value: **N/A**
Description:
Rachel is going to Publix and only has a set amount of money. She wants to spend it wisely by buying as many expensive items as she can until she runs out of money. The items are priced as:

1. Chicken Wings: $13
2. Taquitos: $6
3. Coke: $4
4. Chips: $2
5. Mints: $1

Write a function that asks the user how much money they have with them and calculate what she can buy at the store.

1. Ask the user how much money they have.
2. Calculate how many of each item they can purchase giving priority to the more expensive items.
3. Print out a statement with the calculated number of items on one line, and be sure to add labels to the values so the user knows what the value means.
   (Ex: `With $33 they can buy 2 chicken wings, 1 taquitos, 0 coke, 0 chips, and 1 mints.`)

Hints:

- The modulus (%) and integer division (//) operators will be very useful for this function.

Test Cases (not an exhaustive list):

- `With $33 they can buy 2 chicken wings, 1 taquitos, 0 coke, 0 chips, and 1 mints.)`
- `With $14 they can buy 1 chicken wings, 0 taquitos, 0 coke, 0 chips, and 1 mints`
- `With $26 they can buy 2 chicken wings, 0 taquitos, 0 coke, 0 chips, and 0 mints`
- `With $11 they can buy 0 chicken wings, 1 taquitos, 1 coke, 0 chips, and 1 mints`

Function name (5): **tipCalc**
Parameters: **N/A**
Return value: **N/A**
Description:
Write a function that takes the price of a restaurant meal, asks the user how much they want to tip, and calculates the amount they should leave as tip.

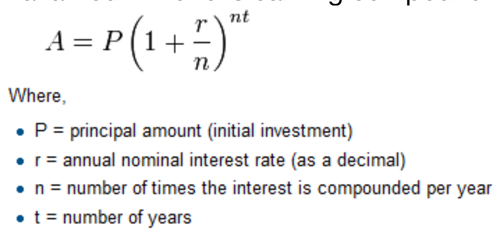
1. Get the price of the meal in dollars and cents from the user; make sure to use a descriptive prompt so the user knows what units to use.

2. Ask the user how many percent they want to tip. If they want to tip 15%, they should enter “15”. (Assume an integer percentage i.e. giving a 15.5% tip is not allowed)
3. Print a descriptive statement of your calculations (see test cases for formatting).

Test Cases (not an exhaustive list):

```
-   If your meal was: $11.0 and you want to give a 18% tip. The tip will be:
    $1.98, so the new total is: $12.98
-   If your meal was: $24.5 and you want to give a 20% tip. The tip will be:
    $4.9, so the new total is: $29.4
-   If your meal was: $30.9 and you want to give a 15% tip. The tip will be:
    $4.63, so the new total is: $35.53
```

Function name (6): **compound_interest**
Parameters: **N/A**
Return value: **N/A**
Description:
The formula for computing the final amount if one is earning compound interest is given as:



$$A = P\left(1 + \frac{r}{n}\right)^{nt}$$

Where,

- P = principal amount (initial investment)
- r = annual nominal interest rate (as a decimal)
- n = number of times the interest is compounded per year
- t = number of years

Calculate the total amount of earning compound interest. Input from the user all the information needed to make your calculations.

Test Cases (not an exhaustive list):

```
-   With an initial investment of $10000.0, an 8.0% annual nominal interest
    rate, compounded 1 times per year after 5 years you will have a final
    amount of: $14693.28.
-   With an initial investment of $1500.0, an 3.0% annual nominal interest
    rate, compounded 1 times per year after 10 years you will have a final
    amount of: $2015.87.
```

Function name (7): **main**
Parameters: **N/A**
Return value: **N/A**
Description:
This function has been started for you. It serves as an introduction for conditional statements (learned later in the course). All you have to do is ask the user what function they will like to call and make the function call yourself.

1. Ask the user to input the number of the function [1-6]
2. Replace the `pass` statements with your function calls.

Grading Rubric

-Mugshot: 30 points
File Named Correctly: 5 Points
Correct File Size: 10 Points
Face Recognizable: 5 Points
- **circleArea**: 10 points
Correct function header: 2 Points
Prints in the correct format: 3 Points
Correct Value is calculated: 5 Points
-**howManyLegs**: 10 points
Correct function header: 2 Points
Prints in the correct format: 3 Points
Correct number of total legs given: 5 Points
-**calcBMI**: 10 points
Correct function header: 2 Points
Prints in the correct format: 3 Points
Correct response is calculated: 5 Points
-**publix**: 10 points
Correct function header: 2 Points
Prints in the correct format: 3 Points
Correct number of items is calculated: 5 Points
-**tipCalc**: 10 points
Correct function header: 2 Points
Prints in the correct format: 3 Points
Correct response is calculated: 5 Points
-**compound_interest:** 15points
Correct function header: 2 Points
Prints in the correct format: 3 Points
Correct value is calculated: 10 Points
-**main**: 5 points
Calls the correct function: 5 Points

Provided

The following file(s) have been provided to you. There are several, but you will only edit one of them:

1. `HW1.py`
   This is the file you will edit and implement. All instructions for what the methods should do are in the docstrings.

Deliverables

You must submit all of the following file(s). Please make sure the filename matches the filename(s) below. Be sure you receive the confirmation email from T-Square, and then download your uploaded files to a new folder and run them.

1. `HW1.py`
2. `First_Last.jpg`