
CMSC 201 Spring 2018

Homework 1 – Pseudocode to Code

Assignment: Homework 1 – Pseudocode to Code

Due Date: Friday, September 14th, 2018 by 8:59:59 PM

Value: 40 points

Collaboration: For Homework 1, collaboration is allowed. Make sure to consult the syllabus about the details of what is and is not allowed when collaborating. You may not work with any students who are not taking CMSC 201 this semester.

If you work with someone, remember to note their name, email address, and what you collaborated on by filling out the Collaboration Log.

You can find the Collaboration Log at <https://tinyurl.com/collab201-fa18>.

Remember that all collaborators need to fill out the log each time; even if the help was only “one way” help.

Make sure that you have a **complete file header comment at the top of each file**, and that all of the information is correctly filled out.

```
# File:      FILENAME.py
# Author:    YOUR NAME
# Date:      THE DATE
# Section:   YOUR DISCUSSION SECTION NUMBER
# E-mail:    YOUR_EMAIL@umbc.edu
# Description:
#           DESCRIPTION OF WHAT THE PROGRAM DOES
```

Instructions

For each of the questions below, you are given a program, expressed in simple pseudocode. (This is similar to the in-class exercises we did during Lecture 2.) From this pseudocode, you must implement a working program in Python. For this exercise, you will only need to use concepts we have discussed in class such as variables, expressions, `input()`, casting to an integer or float, and `print()`.

The pseudocode may combine multiple lines of code into one step, or they may split something that would take a single line of code into multiple pieces. Think carefully about what the overall goal of the program is before you begin coding.

At the end, your Homework 1 files must run without any errors.

Additional Instructions – Creating the hw1 Directory

During the semester, you'll want to keep your different Python programs organized, organizing them in appropriately named folders (also known as directories).

For Homework 1, you can store all four of the files you'll be creating in a single directory. First, navigate to the **Homeworks** directory inside the **201** directory (you can do this in a single "cd" command, as shown below). Next, create a directory to hold your Homework 1 files, and finally go into it.

```
linux3[1]% cd 201/Homeworks
linux3[3]% mkdir hw1
linux3[4]% cd hw1
linux3[5]% █
```

From here, you can use `emacs` to start creating and writing your different Homework 1 Python programs.

You don't need to make a separate folder for each file. You should store all of the Homework 1 files in the same `hw1` folder.

Coding Standards

Prior to this assignment, you should read the Coding Standards, linked on the course website at the top of the “Assignments” page.

For now, you should pay special attention to the sections about:

- Naming Conventions
- Use of Whitespace
- Comments (specifically, File Header Comments)

We will not grade “harshly” on Coding Standards this early in the semester, but you should start forming good habits now. Make sure to pay attention to your TA’s feedback when you receive your Homework 1 grade back.

Additional Specifications

For this assignment, you do not need to worry about any “input validation.”

If the user enters a different type of data than what you asked for, your program may crash. This is acceptable.

If the user enters “bogus” data (for example: a negative value when asked for a positive number), this is acceptable. (Your program does not need to worry about correcting the value or fixing it in any way.)

For example, if your program asks the user to enter a whole number, it is acceptable if your program crashes if they enter something else like “dog” or “twenty” or “88.2” instead.

Here is what that error might look like:

```
Please enter a number: twenty
Traceback (most recent call last):
  File "test_file.py", line 10, in <module>
    num = int(input("Please enter a number: "))
ValueError: invalid literal for int() with base 10: 'twenty'
```

Questions

Each question is worth 4 points. Following the directions is worth 8 points.

Question 1:

Write your program for Question 1 in a file called `hw1_part1.py`.

You stumbled upon a treasure chest while out on a walk. This program, shown in pseudocode, prints out information about how many gems you found in your treasure chest.

Translate this pseudocode into a Python program.

```
Set a number value for a variable called numEmeralds
Set a number value for a variable called numSapphires
Set a number value for a variable called numDiamonds
```

Using those variables:

```
Print out how many emeralds, sapphires, and diamonds
    you have in your treasure chest
Print out the total number of gems in your chest
```

Here is some sample output, using sample values for a chest that Prof. Neary found last week. (The number of emeralds, sapphires, and diamonds you have will probably be different.)

(Yours does not have to match this word for word, but it should be similar.)

```
linux3[11]# python3 hw1_part1.py
I have 6 emeralds in my chest
I have 2 sapphires in my chest
I have 11 diamonds in my chest
I found 19 gems in my chest.
```

Question 2

Write your program for Question 2 in a file called `hw1_part2.py`.

This program, shown in pseudocode, asks for information about how much the user's weekly allowance is, the cost to buy a new phone, and then calculates and prints out the number of weeks they'll need to save their allowance before being able to afford the sticker price of the phone.

Translate this pseudocode into a Python program.

```
Ask the user how much their weekly allowance is
Ask the user how much the new phone costs
Calculate the number of weeks they'll need to save up to
afford the phone, and print it
```

For this program, the names of the variables are not given to you. You should choose meaningful variable names.

Here is some sample output, with the user input in **blue**.

(Yours does not have to match this word for word, but it should be similar. The calculated costs should be the exact same, given the same input.)

```
linux3[196]% python3 hw1_part2.py
Enter your weekly allowance: 11.25
Enter the cost of a new phone: 75.00
It will take 6.666666666666667 weeks.

linux3[197]% python3 hw1_part2.py
Enter your weekly allowance: 23.50
Enter the cost of a new phone: 399.00
It will take 16.97872340425532 weeks.

linux3[198]% python3 hw1_part2.py
Enter your weekly allowance: 15
Enter the cost of a new phone: 255
It will take 17.0 weeks to get that phone.
```

Question 3

Write your program for Question 3 in a file called `hw1_part3.py`.

This program, shown in pseudocode, asks the user for the name of their favorite food, and then prints out that the food is delicious.

Translate this pseudocode into a Python program.

Ask the user for the name of their favorite food and store it in a variable called `favFood`
Print out that `favFood` is delicious

Here is some sample output, with the user input in blue.

(Yours does not have to match this word for word, but it should be similar.)

```
linux3[216]% python3 hw1_part3.py
Enter your favorite food: pizza
pizza is so delicious!

linux3[217]% python3 hw1_part3.py
Enter your favorite food: popcorn
popcorn is so delicious!

linux3[218]% python3 hw1_part3.py
Enter your favorite food: Hershey's Milk Chocolate
Hershey's Milk Chocolate is so delicious!

linux3[219]%
```

Question 4

Write your program for Question 4 in a file called `hw1_part4.py`.

This program, shown in pseudocode, asks the user for information about a movie they saw in theaters, and calculates and prints out the total cost of that movie.

Translate this pseudocode into a Python program.

```
Ask the user the cost of the movie ticket
Ask the user the cost of the popcorn
Ask the user the cost of the drink
Ask the user the cost of the candy
Calculate and print the total cost of the movie
```

For this program, the names of the variables are not given to you. You should choose meaningful variable names.

Here is some sample output, with the user input in **blue**.

(Yours does not have to match this word for word, but it should be similar. The calculated costs should be the exact same, given the same input.)

```
linux3[22]% python3 hw1_part4.py
Enter the ticket price: 11.75
Enter the popcorn price: 5.75
Enter the drink price: 4.35
Enter the candy price: 6.45
The movie had a total cost of 28.3

linux3[23]% python3 hw1_part4.py
Enter the ticket price: 10.45
Enter the popcorn price: 3.85
Enter the drink price: 2.45
Enter the candy price: 0
The movie had a total cost of 16.75
```

Question 5

Write your program for Question 5 in a file called `hw1_part5.py`.

This program, shown in pseudocode, asks the user for information about an Olympian, and prints out a summary of that information.

Translate this pseudocode into a Python program.

Ask the user what the person's name is, and store it in a variable called `playerName`

Ask the user for what sport the person plays, and store it in a variable called `sportInfo`

Ask the user for the medal they won in that sport, and store it in a variable called `medalWon`

Print out that `playerName` won the `medalWon` medal in `sportInfo`

Here is some sample output, with the user input in **blue**.

(Yours does not have to match this word for word, but it should be similar.)

```
linux3[34]% python3 hw1_part5.py
Enter the player's name: Michael Phelps
Enter the sport they play: swimming
Enter the medal they won: gold
The Olympian Michael Phelps won a gold medal playing
the sport swimming

linux3[34]% python3 hw1_part5.py
Enter the player's name: Apolo Ohno
Enter the sport they play: speed skating
Enter the medal they won: silver
The Olympian Apolo Ohno won a silver medal playing the
sport speed skating
```


Question 6

Write your program for Question 6 in a file called `hw1_part6.py`.

This program, shown in pseudocode, asks the user for information about a bill they've received, and calculates and prints out how much they should pay in total.

Translate this pseudocode into a Python program.

```
Ask the user how much the bill was
Ask the user what the tax percentage is
Ask the user what percentage they want the tip to be
Calculate and print the total cost of the bill
```

For this program, the names of the variables are not given to you. You should choose meaningful variable names.

Here is some sample output, with the user input in **blue**.

(Yours does not have to match this word for word, but it should be similar. The calculated cost should be the exact same, given the same input.)

```
linux3[34]% python3 hw1_part6.py
How much was the bill? 100
How much is tax in your area? 6
What percentage do you want to tip? 15
The total cost of the meal will be 121.0

linux3[35]% python3 hw1_part6.py
How much was the bill? 26.88
How much is tax in your area? 4.5
What percentage do you want to tip? 16.7
The total cost of the meal will be 32.578559999999996
```

Question 7

Write your program for Question 7 in a file called `hw1_part7.py`.

This program, shown in pseudocode, asks the user for the rating a Lyft driver has earned, and then computes the average rating for that driver.

Translate this pseudocode into a Python program.

```
Ask the user how their driving was (from 1 to 5), and
store it in a variable called drivingRating
Ask the user how their friendliness was (from 1 to 5),
and store it in a variable called friendRating
Ask the user how the music played was (from 1 to 5), and
store it in a variable called musicRating
```

Compute and print the average of the three ratings

Here is some sample output, with the user input in **blue**.

(Yours does not have to match this word for word, but it should be similar. The calculated ratings should be the exact same, given the same input.)

```
linux3[38]% python3 hw1_part7.py
How would you rate their driving? (1-5): 4
How would you rate their friendliness? (1-5): 3
How would you rate their music? (1-5): 2
Your overall experience was 3.0 stars.

linux3[39]% python3 hw1_part7.py
How would you rate their driving? (1-5): 5
How would you rate their friendliness? (1-5): 4
How would you rate their music? (1-5): 5
Your overall experience was 4.666666666666667 stars.
```

Question 8

Write your program for Question 8 in a file called `hw1_part8.py`.

You got a \$50 gift card to use at your local pizza place. This program, shown in pseudocode, asks the user for information about how many items of food they want to buy at a fixed price, and calculates and prints out the total cost of the meal as well as how much is left on the gift card. Translate this pseudocode into a Python program:

```
Use a variable to set the value of your gift card to 50
Use variables to set number values for the cost of pizza
slices, burgers, and cookies
```

```
Ask the user how many pizza slices, burgers, and cookies
they want to order
```

```
Calculate and print the total cost of the meal
```

```
Calculate and print how much money you have left over
```

For this program, the names of the variables are not given to you. You should choose meaningful variable names.

Here is some sample output, using sample cost values for *my* example restaurant. (The price of your pizza slices, hamburgers, and cookies should be different.) The user input is shown in **blue**.

(Yours does not have to match this word for word, but it should be similar. The calculated costs should be the exact same, given the same input and dish costs.)

```
bash-4.1$ python hw1_part8.py
Pizza slices cost 3.5
How many pizza slices would you like to order? 3
Burgers cost 6.8
How many burgers would you like to order? 1
Cookies cost 0.99
How many cookies would you like to order? 5

The total cost of your meal will be 22.25
You have 27.75 left on the gift card.
```

Submitting

NOTE: How to submit is covered in detail in Lab 1 and Homework 0. If you have not read those assignments yet, you should do so before completing this part of the homework.

Once your `hw1_part1.py`, `hw1_part2.py`, `hw1_part3.py`, `hw1_part4.py`, `hw1_part5.py`, `hw1_part6.py`, `hw1_part7.py`, and `hw1_part8.py` files are complete, it is time to turn them in with the `submit` command. (You may also turn in individual files as you complete them. To do so, only `submit` those files that are complete.)

You must be logged into your account on GL, and you must be in the same directory as your Homework 1 Python files. To double-check you are in the directory with the correct files, you can type `ls`.

```
linux1[3]% ls
hw1_part1.py  hw1_part3.py  hw1_part5.py  hw1_part7.py
hw1_part2.py  hw1_part4.py  hw1_part6.py  hw1_part8.py
linux1[4]% █
```

(continued on next page)

To submit your Homework 1 Python files, we use the `submit` command, where the class is `cs201`, and the assignment is `HW1`. Type in (all on one line) `submit cs201 HW1 hw1_part1.py hw1_part2.py hw1_part3.py hw1_part4.py hw1_part5.py hw1_part6.py hw1_part7.py hw1_part8.py` and press enter.

```
linux1[4]% submit cs201 HW1 hw1_part1.py hw1_part2.py
hw1_part3.py hw1_part4.py hw1_part5.py hw1_part6.py
hw1_part7.py hw1_part8.py
Submitting hw1_part1.py...OK
Submitting hw1_part2.py...OK
Submitting hw1_part3.py...OK
Submitting hw1_part4.py...OK
Submitting hw1_part5.py...OK
Submitting hw1_part6.py...OK
Submitting hw1_part7.py...OK
Submitting hw1_part8.py...OK
linux1[5]% █
```

If you don't get a confirmation like the one above, check that you have not made any typos or errors in the command.

You can check that your homework was submitted by following the directions in Homework 0. Double-check that you submitted your homework correctly, since **an empty file will result in a grade of zero for this assignment.**