CSE 231 Spring 2016

Programming Project #3

Assignment Overview
This assignment involves coding and testing of a program using Python control (*if, for, while*) from Chapter 2.

This assignment is worth 20 points (2% of course grade), and must be completed before 11:59 PM on Monday, February 1.

Assignment Deliverable

The deliverable for this assignment is the following file:

`proj03.py` – the source code for your Python program

Be sure to use the specified file name and to submit it for grading via the **handin** system before the project deadline.

Background

You are going to write a program to help young students practice their addition. It will present them with random numbers to add together, get their answer and report if it is correct or not. They'll get a report at the end of how many problems they got right. The user will be able to specify how difficult the problems will be.

Program Specifications

1. Your program will prompt for two numbers:
   A. Difficulty (an integer >= 2).
   B. Number of problems (an integer >= 1)
   C. Error checking: you will check that the difficulty is greater than or equal to 2 and that the number of problems specified is greater than or equal to 1. If an error is found, print an appropriate error message and end the program without drawing any figures. Do your error checking after *both* values are input. The error message must be specific to the error so that if one input has an error the message will be different than if both have an error—see samples below. (Assume that the user will input an integer; if something other than an integer is input, it is fine for Python to halt your program with an error—we will learn a good way to handle that later.)
2. Your program will generate addition problems of the specified difficulty, prompt for an answer, provide feedback on whether the answer is correct or not (and provide the correct sum, if the answer was incorrect), and keep track of the number of correct answers. "Difficulty" determines two things:
   i. The range of digits in each number, e.g. difficulty of 3 means that the addition will be numbers up to 3 digits (i.e. 0 – 999).
   ii. The number of values being added, e.g. difficulty of 3 means that 3 numbers will be added, e.g. x + y + z where each number has up to 3 digits.

3. After all the problems have been solved you will provide feedback of the number of correct answers out of the total possible and provide the percent correct. The percent should be rounded to the nearest tenth (use the `round` function described below).

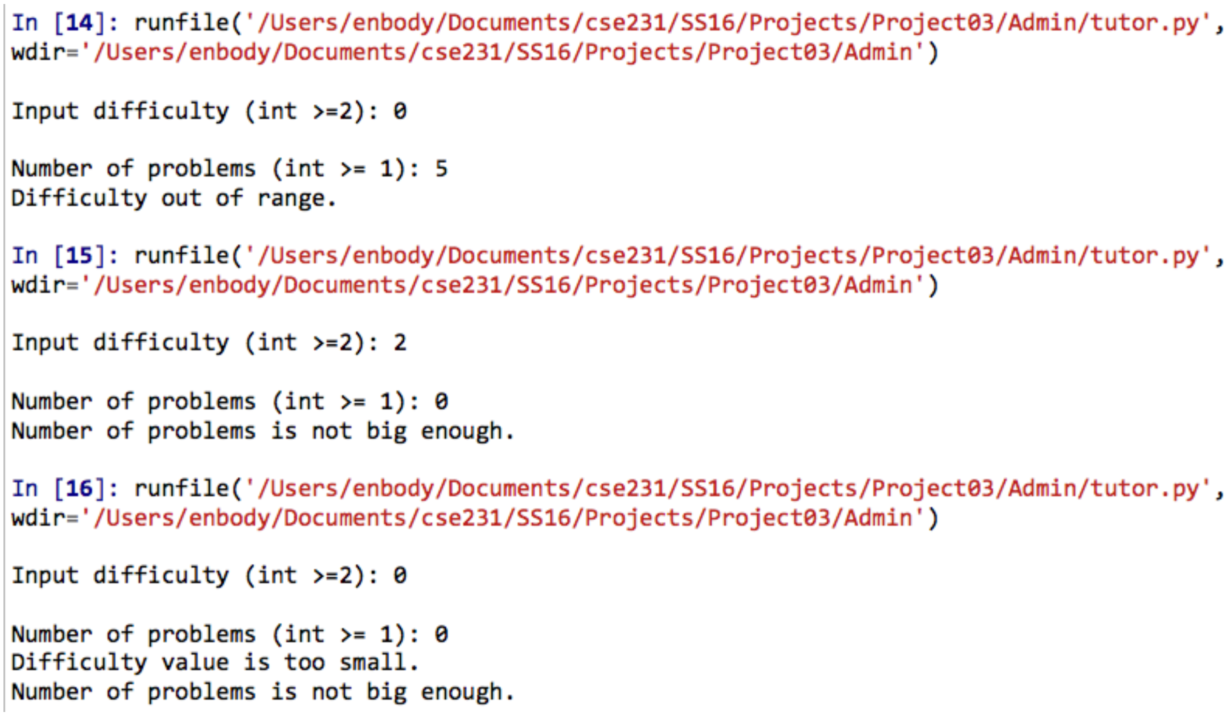
Assignment Notes:

1. Follow the Coding Standards 1-7.
2. Python has a `round()` function that will round a floating-point value to a specified number of places. For example if `n = 12.357`, `round(n,2)` returns `12.36`. Experiment with the `round` function in the IPython shell to better understand how it works.
3. Remember to `import random`. For this project you want random integers so use the `random.randint()` function. The `randint(a, b)` function returns a random integer in the range [a, b], including both end points. Experiment with the `randint` function in the IPython shell to better understand how it works.

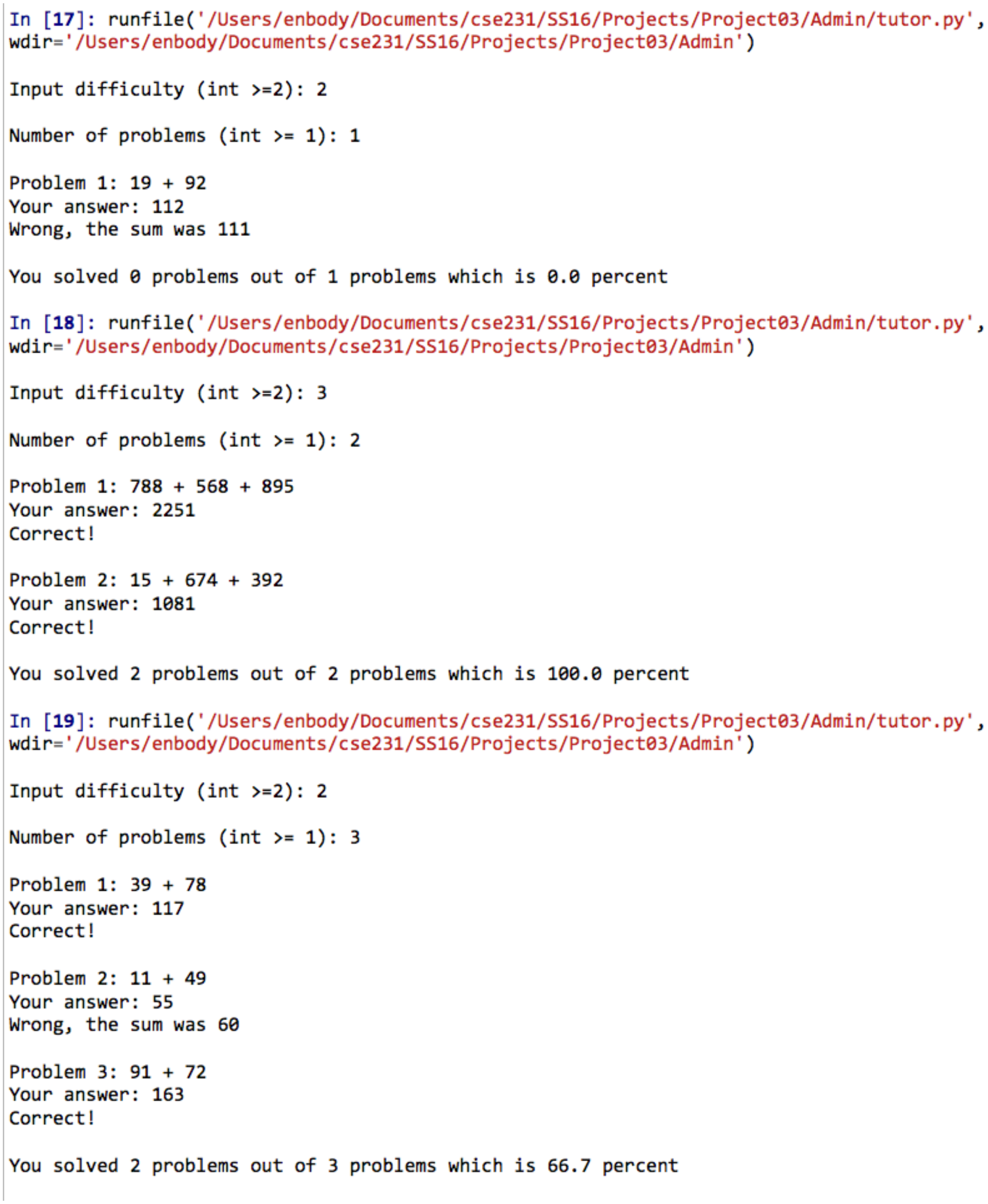
Getting Started

1. Divide-and-conquer is an effective problem solving strategy. Consider taking these steps in designing your program. Get each step working before moving on to the next. Start a new program named `proj03.py`.
   a. Prompt for the number of problems and print out “Problem 1”, “Problem 2”,… Hint: use `for` (because you know how many times to print out “Problem X”).
   b. Prompt for the difficulty. Let’s say that the difficulty entered is 3. For each problem print out that many (3) random numbers each in the range of 0-999.
   c. Print out the random numbers for each problem on one line. Next, place a “+” sign between each number.
   d. Find the sum of the numbers.
   e. Prompt for an answer.
   f. Check the answer and provide appropriate feedback. Hint: use `if-else`
   g. Provide a summary of results at the end.
   h. Add error checking as specified above.
2. Test your program. Begin using values from the Sample Interaction, but do more tests than that.

Sample Interaction



```
In [14]: runfile('/Users/enbody/Documents/cse231/SS16/Projects/Project03/Admin/tutor.py',
wdir='/Users/enbody/Documents/cse231/SS16/Projects/Project03/Admin')

Input difficulty (int >=2): 0

Number of problems (int >= 1): 5
Difficulty out of range.

In [15]: runfile('/Users/enbody/Documents/cse231/SS16/Projects/Project03/Admin/tutor.py',
wdir='/Users/enbody/Documents/cse231/SS16/Projects/Project03/Admin')

Input difficulty (int >=2): 2

Number of problems (int >= 1): 0
Number of problems is not big enough.

In [16]: runfile('/Users/enbody/Documents/cse231/SS16/Projects/Project03/Admin/tutor.py',
wdir='/Users/enbody/Documents/cse231/SS16/Projects/Project03/Admin')

Input difficulty (int >=2): 0

Number of problems (int >= 1): 0
Difficulty value is too small.
Number of problems is not big enough.
```



```
In [17]: runfile('/Users/enbody/Documents/cse231/SS16/Projects/Project03/Admin/tutor.py',
wdir='/Users/enbody/Documents/cse231/SS16/Projects/Project03/Admin')

Input difficulty (int >=2): 2

Number of problems (int >= 1): 1

Problem 1: 19 + 92
Your answer: 112
Wrong, the sum was 111

You solved 0 problems out of 1 problems which is 0.0 percent

In [18]: runfile('/Users/enbody/Documents/cse231/SS16/Projects/Project03/Admin/tutor.py',
wdir='/Users/enbody/Documents/cse231/SS16/Projects/Project03/Admin')

Input difficulty (int >=2): 3

Number of problems (int >= 1): 2

Problem 1: 788 + 568 + 895
Your answer: 2251
Correct!

Problem 2: 15 + 674 + 392
Your answer: 1081
Correct!

You solved 2 problems out of 2 problems which is 100.0 percent

In [19]: runfile('/Users/enbody/Documents/cse231/SS16/Projects/Project03/Admin/tutor.py',
wdir='/Users/enbody/Documents/cse231/SS16/Projects/Project03/Admin')

Input difficulty (int >=2): 2

Number of problems (int >= 1): 3

Problem 1: 39 + 78
Your answer: 117
Correct!

Problem 2: 11 + 49
Your answer: 55
Wrong, the sum was 60

Problem 3: 91 + 72
Your answer: 163
Correct!

You solved 2 problems out of 3 problems which is 66.7 percent
```