



# Assignment 2

## Functions

Date Due: September 27, 2019, 6:00pm

Total Marks: 31

### General Instructions

- **This assignment is individual work.** You may discuss questions and problems with anyone, but the work you hand in for this assignment must be your own work.
- Each question indicates what to hand in. You must give your document the name we prescribe for each question, usually in the form **aNqM**, meaning Assignment N, Question M. Put your name and student number at the top of every document you hand in. These conventions assist the markers in their work. Failure to follow these conventions will result in needless effort by the markers, and a deduction of grades for you. Do not submit folders, zip documents, even if you think it will help.
- Programs must be written in Python, and the file format must be text-only, with the file extension `.py`.
- Documents submitted for discussion questions should make use of common file formats, such as plain text (`.txt`), Rich Text (`.rtf`), and PDF (`.pdf`). We permit only these formats to ensure that our markers can open your files conveniently.
- **Assignments must be submitted electronically to Moodle.** There is a link on the course webpage that shows you how to do this. Your TAs will cover this in the first tutorial.
- **Moodle will not let you submit work after the assignment deadline.** It is advisable to hand in each answer that you are happy with as you go. You can always revise and resubmit as many times as you like before the deadline; only your most recent submission will be graded.
- Questions are annotated using descriptors like "easy" "moderate" and "tricky". All students should be able to obtain perfect grades on "easy" problems. Most students should obtain perfect grades on "moderate" problems. The problems marked "tricky" may require significantly more time, and only the top students should expect to get perfect grades on these. We use these annotations to help students be aware of the differences, and also to help students allocate their time wisely. Partial credit will be given as appropriate, so hand in anything you've done, even if it's not perfect.

**Question 1 (9 points):**

**Purpose:** To distinguish between the different parts of functions and function calls.

**Degree of Difficulty:** Easy

Below is a program that uses functions to compute the total cost of purchasing coffees. It includes the ability to calculate tax and subtract a discount from the purchase.

```
1 def calculate_coffee( num_coffee, price_coffee, discount):
2     """
3     Computes the total amount due for the purchase of num_coffee
4
5     num_coffee: An Integer for the number of coffees purchased
6     price_coffee: A Float for the price per coffee
7     discount: A Float for a discount off the purchase price
8     returns: the total amount due from the customer
9     """
10    total_before_tax = num_coffee * price_coffee
11    total_after_tax = add_tax(total_before_tax)
12    total_after_discount = total_after_tax - discount
13    return total_after_discount
14
15 def add_tax(cost):
16     """
17     Adds 7% tax to the cost
18
19     returns total the amount including tax
20     """
21    total = cost * 1.07
22    return total
23
24 # compute the total for 1 coffee at $2.00 per coffee, no discount
25 customer1 = calculate_coffee(1, 2.00, 0)
26
27 # compute the total for 3 coffees at $3.25 per coffee, $2.00 discount
28 customer2 = calculate_coffee(3, 3.25, 2.00)
29
30 # compute the total for 7 coffees at $2.75 per coffee, $5.00 discount
31 customer3 = calculate_coffee(7, 2.75, 5.00)
```

Answer the following questions. Refer to line numbers if you find it helpful in your answers.

- List all of the *function parameters* in this program.
- List all of the *function arguments* in this program.
- List all of the variables that are not *arguments* or *parameters*.
- What is the *scope* of the following variables:
  - total\_after\_tax
  - cost
  - customer2
- How many *function definitions* are in this program?
- How many *function calls* are in this program?
- What is the value of `customer1` when the program ends?

## What to Hand In

Hand in your answers in a file called a2q1. Allowed file formats are plain text (.txt), Rich Text (.rtf), and PDF (.pdf).

## Evaluation

- 1 mark for each of parts (a) through (c). Full marks if completely correct, 0.5 marks if partially correct, 0 marks if nothing correct;
- 3 marks for part (d), one for each scope correctly identified;
- 1 mark each for parts (e) and (f);
- 1 mark for part (g) for variable value correctly identified.

## Question 2 (5 points):

**Purpose:** To write a function that performs a subtask and returns an answer.

**Degree of Difficulty:** Easy

A travel agent would like to compute the total cost of a trip that includes a flight, plus a number of nights at a hotel for a group of people. The agent is assuming double occupancy, which means two will share each hotel room, unless there is an odd number of people. Thus to determine the **number of rooms** required each night you could use the formula

$$\text{number\_of\_rooms} = \text{number\_of\_people} // 2 + \text{number\_of\_people} \% 2$$

Write a Python function `trip_cost` that takes four parameters as input and returns the **total cost of the trip**. The four parameters should be the **airfare** in dollars, the **room cost** per night in dollars, the **number of people**, and the **number of nights**.

Ask the user to input values for each of the function arguments from the console. You may assume that the user supplies valid input. Call the `trip_cost` function with the arguments you read from the console and output the its return value to the console.

Your function `trip_cost` should do no console input and no console output. It must receive its input through its parameters and send its output using a return value.

**NOTE:** The cost of flight is **per person** and not for the whole group.

## Sample Run

Here is an example of how your program's console output might look. Green text was entered by the user; blue text came from data returned by the function.

```
Calculate Trip Cost
Enter cost of flight ($): 1000
Enter cost of a double room per night:100.00
Enter the number of people: 11
Enter the number of nights:7
The total cost of the trip for the group is $15200.0
```

## What to Hand In

Hand in your solution in a file called `a2q2.py`.

## Evaluation

- 1 mark for reading inputs from the console.
- 3 marks for the `trip_cost` function having a correct header, a reasonable docstring, and correct return value.
- 1 mark for printing function output to console.

### Question 3 (4 points):

**Purpose:** To write a function that performs a subtask and returns an answer.

**Degree of Difficulty:** Easy

In Physics the speed of a moving object can be calculated by dividing the distance traveled by the time taken:

$$speed = distance \div time$$

Write a Python function named `calculate_speed` which has two parameters (distance and time) and returns the value of speed.

Also write a main program that reads the two necessary values from the user, passes the entered values to the `calculate_speed` function to perform the computation, obtains the function's return value, then prints a message to the console that reports the value of speed.

Your function `calculate_speed` should do no console input and no console output. It must receive its input through its parameters and send its output using a return value.

### Sample Run

Here is an example of how your program's console output might look. Green text was entered by the user; blue text came from data returned by the `calculate_speed` function.

```
Calculate Speed
Enter desired distance (km): 510
Enter desired time (hours): 5
In order to travel 510 km in 5 hours you must travel at 102 km/h.
```

The above shows the result of traveling 510 km over 5 hours.

### What to Hand In

Hand in your solution in a file called `a2q3.py`.

### Evaluation

- 1 mark for correct function header (name and parameters)
- 1 mark for appropriate docstring
- 1 mark for returning correct value
- 1 mark for correct console input

## Question 4 (13 points):

**Purpose:** Solve a problem by writing multiple functions that perform different subtasks, and combining their use to solve a larger problem. Also to practice documenting functions.

**Degree of Difficulty:** **Moderate**

Siblings John and Jane opened up a lemonade stand for a day to raise funds for buying their first dog. The lemonade was a hit! What they weren't sure about was how much money they earned at the end of the day.

Your task is to write a Python program that calculates and displays the amount of funds raised by the youth after taking their expenses into consideration, that is, their profit. To accomplish this, you will need to write three Python functions: `cost`, `revenue`, and `profit`.

The `cost` function should take parameters `n_lemons` (the number of lemons bought and used to make lemonades), `cost_per_lemon` (price per lemon), and return the total cost of running the lemonade service. The total cost is:

$$total\_cost = number\ of\ lemons \times cost\ per\ lemon.$$

The `revenue` function should take parameters `donations` (the amount of money donated towards their cause), `n_lemonades` (the number of lemonades sold), `price_per_lemonade` (the selling price for one lemonade) and return the total amount of money earned. The total money earned is:

$$total\_revenue = donations + money\ earned\ from\ lemonades\ sold$$

The `profit` function determines the net profit (net money earned) by the youth. It should:

- call `cost` and `revenue` and use their return values to compute the net profit earned by the siblings;
- return the computed profit.

The parameters of the `profit` function should be chosen so that they provide the necessary information required to call the `cost` and `revenue` functions. The profit is:

$$profit = total\_revenue - total\_cost$$

You must now test your program by preparing three examples (of number of lemons used, buying price per lemon, number of lemonades sold, selling price per lemonade, and donations) in advance so that you can judge whether your program is correct. Write code to ask the user for all five of these inputs, then display the cost, revenue, and profit of each prepared example to the console. Copy the console output from each of your three testing example into a text file and hand that text file in (see "What to Hand In", below).

You may assume that the user supplies valid input from the console, that is, a positive number for the number of lemons, the buying price per lemon, the number of lemonades sold, the selling price per lemonade, and for monetary amount of donations received. See the sample run on the next page.



## Sample Run

Here is an example of how your program's console output might look. **Green** text was entered by the user; **blue** text came from data returned by the function.

```
How many lemons were used?: 5
What was the cost per lemon (in dollars)?: 1.15
How many lemonades were sold?: 25
What was the selling price of one lemonade (in dollars)?: 0.50
How much money did you receive in donations (in dollars)?: 10.00
Revenue: $22.5
Costs: $5.75
Profit: $16.75
```

## What to Hand In

- Your Python functions in a file called `a2q4.py`.
- Console output of your three examples in a text file called `a2q4_testing`. Allowed file formats are plain text (.txt), Rich Text (.rtf), and PDF (.pdf).

## Evaluation

- 1 mark for reading inputs from the console.
- 9 marks for each function having an accurate header, a reasonable docstring, and correct return value (3 marks per function).
- 3 marks for the document containing the console outputs from running your program on your three test cases.