

PHY494 Solution 03

January 24, 2019

Copyright © 2016–2019 Ian Kenney, Oliver Beckstein. ALL RIGHTS RESERVED. These solutions can ONLY be used by current students of the ASU PHY494 Class “Computational Methods in Physics”. DISSEMINATION IN ANY FORM IS PROHIBITED.

(See *assignment_02.pdf* for the detailed break-down of points.)

2.1 Operators (6 points)

a) Output:

```
15.0 0.5 9765625.0 10.0
```

The operations are addition, division, exponentiation, and remainder after division.

b) Error: `ZeroDivisionError: float division by zero`

Python is letting the user know that a division by zero occurred. The reason is that `y//x` performs floor division and yields `-1.0`, which then leads to the whole denominator being zero.

2.2 Very Simple Temperature Calculator (13 points)

a) BONUS: The difference in Kelvin is

$$\begin{aligned}\Delta T &= T_2 - T_1 = \\ &= \left[\frac{5}{9}(\theta_2 - 32) + 273.15 \right] - \left[\frac{5}{9}(\theta_1 - 32) + 273.15 \right] = \\ &= \frac{5}{9}(\theta_2 - \theta_1) = \frac{5}{9}\Delta\theta\end{aligned}$$

- b) Sum of T in K and $\Delta\theta$ in Fahrenheit

$$T' = T + \frac{5}{9}\Delta\theta$$

gives the total T' in Kelvin.

- c) Program `addtemperatures.py`

```
# HW02 2.2 addtemperatures.py

T_K = float(input("Temperature in Kelvin --> "))
theta_F = float(input("Temperature difference in Fahrenheit --> "))

delta_T_K = 5/9 * theta_F
T_total = T_K + delta_T_K

print("Total T " + str(T_total) + " K")
```

- d) Input and output:

```
Temperature in Kelvin --> 265
Temperature difference in Fahrenheit --> 63
Total T 300.0 K
```

(Correct sum is 300 K.)

2.3 Version control with Git (9 points)

- a) Briefly explain what a version control system such as Git does and how it can help you. (For your answer, it is sufficient to focus on three different aspects out of many — choose the ones that you find most important.)
1. A version control system keeps track of the whole history of changes to a set of files.
 2. It allows accessing any point in the history (like an undo).
 3. It also supports combining changes from different sources.
- b) `git init` initializes an empty repository whereas `git clone` initializes an empty repository and then immediately fills it with the history from a “remote” repository.
- c) `git add` adds a file or changes to a file to the staging area and marks it to be committed. `git commit` then takes the contents from the staging area and adds them to the repository history. (You need to commit to record what changed.)
- d) `git push` inserts the local history into a remote history but does not add new history to the local repository (which is what `git commit` does).

2.4 Be the Git (10 points)

Your solution should be in form of a table like the following:

command	staging area	repository
git init		
git add shopping.txt	shopping.txt	
git add planets/alderaan.jpg	shopping.txt planets/alderaan.jpg	
git commit		shopping.txt planets/alderaan.jpg
git add vehicles/stations	vehicles/stations/deathstar.xxx	shopping.txt planets/alderaan.jpg
git commit		shopping.txt planets/alderaan.jpg vehicles/stations/deathstar.xxx
git add vehicles TODO	vehicles/ships/destroyer.txt vehicles/ships/interceptor.txt TODO	shopping.txt planets/alderaan.jpg vehicles/stations/deathstar.xxx
git commit		shopping.txt planets/alderaan.jpg vehicles/stations/deathstar.xxx vehicles/ships/destroyer.txt vehicles/ships/interceptor.txt TODO
git rm planets/alderaan.jpg	planets/alderaan.jpg (delete)	shopping.txt planets/alderaan.jpg vehicles/stations/deathstar.xxx vehicles/ships/destroyer.txt vehicles/ships/interceptor.txt TODO
git commit		shopping.txt vehicles/stations/deathstar.xxx

command	staging area	repository
		vehicles/ships/destroyer.txt vehicles/ships/interceptor.txt TODO
git add planets	planets/hoth.jpg planets/tatooine.jpg	shopping.txt vehicles/stations/deathstar.xxx vehicles/ships/destroyer.txt vehicles/ships/interceptor.txt TODO
git commit		shopping.txt vehicles/stations/deathstar.xxx vehicles/ships/destroyer.txt vehicles/ships/interceptor.txt TODO planets/hoth.jpg planets/tatooine.jpg

Notes:

- `jedi_locations.map` is never under version control in this sequence.
- use :

`git status`

to see files in the staging area (“new file” or “modified” in green).

When you commit, note which files are created or removed.

- The Solution also contains an example file tree in the directory **Documents**, which you can use for testing. However, it is not a good idea to create a git repository inside another git repository — much confusion will ensue. Therefore, if you want to play with the **Documents** tree we suggest you do the following and create a temporary directory (assuming that you are located in the **02/Solutions** directory):

```
mkdir ~/tmp
cp -r Documents ~/tmp
```

```
cd ~/tmp/Documents
git init
git add shopping.txt
git add planets/alderaan.jpg
# ...
```

When you are done, you can just (careful, we have to use `rm -rf!`) remove the `~/tmp/Documents` directory again:

```
cd
rm -rf ~/tmp/Documents
```

This will remove the `~/tmp/Documents` directory including the git repository that you created inside it.

2.5 Your GitHub account (10 points)

Write your GitHub username, for example, *dvader*.