CS3723 Program 5 Python (50 pts)

We are receiving score information from a feed which prefers to enhance it with splashy words. You will read an input file in the format

TEAMS

*team nickname1, nickname2, ...*

*...*

ENDTEAMS

WEEK *weekNr*

*team splashy team, score reference*

*...*

ENDWEEK

WEEK *weekNr*

*team splashy team, score reference*

*...*

ENDWEEK

**Example Data:**

TEAMS

UTSA ROADRUNNERS, RUNNERS

NT MEAN GREEN, GREEN

RICE OWLS

UTEP MINERS, GOLD MINERS

MISS GOLD EAGLES, EAGLES

MARS THUNDERING HERD, HERD, MARTIANS

ENDTEAMS

WEEK 1

RUNNERS UPSET RICE, 10 TO 3

GREEN SLAUGHTERED BY EAGLES, 6 TO 48

MINERS DEFEAT MARTIANS, 12 TO 7

ENDWEEK

WEEK 2

UTSA TOPS MINERS, 14 TO 13

EAGLES SOAR OVER OWLS, 21 TO 14

MARS LOSES TO MEAN GREEN, 21 - 13

ENDWEEK

WEEK 3

UTEP BEAT BY EAGLES, 23 TO 24

UTSA SHUTS OUT HERD, 54 TO 0

OWLS SLAUGHTERED BY MEAN GREEN, 36 TO 0

ENDWEEK

Most of the time the score reference is in order by the first team. **It is sometimes reversed**. (reality stinks)

When teams are referenced, the name might be the team name or one of many nicknames. Some nicknames are single words (e.g., "EAGLES", "MINERS", "RUNNERS", "ROADRUNNERS"), but some are two words (e.g., "MEAN GREEN", "GOLD EAGLES", "GOLD MINERS")

The *splashy* can contain words which mean a win for the first team or a loss by the first team:

win:

**"BEAT"**, **"BEATS"**,   
 **"DEFEAT"**, **"DEFEATS"**, **"DEFEATED"**,  
 **"SLAUGHTER"**, **"SLAUGHTERS"**,**"SLAUGHTERED"**,  
 **"WHIP"**, **"WHIPS"**, **"WHIPPED"**,  
 **"TOP"**, **"TOPS"**, **"TOPPED"**,  
 **"UPSET"**, **"UPSETS"**,  
 **"SHUT"**, **"SHUTS"**, **"SHUTS OUT"**  
 **"EAT"**, **"EATS"**,  
 **"DESTROY"**, **"DESTROYS"**, "**DESTROYED**",   
 **"SOAR"**, **"SOARS", "SOAR OVER", "SOARS OVER"**

loss:

**"LOSES TO", "LOST TO",**

**"BEAT BY"**, **"BEATEN BY"**,   
 **"DEFEATED BY"**,  
 "**SLAUGHTERED BY"**,  
 **"WHIPPED BY"**,  
 **"TOPPED BY"**,  
 **"UPSET BY"**,   
 **"EATEN BY"**,   
 **"DESTROYED BY"**

Notes:

1. Launch your program using:

python3 p5driver.py p5Input.txt > p5out.txt

1. Notice that some verbs when followed by the prepositions "TO" or "BY" change from a win for the first team to a loss by the first team:
   * RUNNERS BEAT OWLS
   * RUNNERS BEAT BY OWLS
2. **Only the words listed should be recognized as verbs**. There will be additional unknown verbs in the data feed. In these cases, use the score to determine the winner, assuming the first score is for the first team and the second score is for the second team. Do NOT add those additional unknown words to your code. Show one of these messages:
   * Unknown verb, by score, assume team 1 won
   * Unknown verb, by score, assume team 2 won
3. After reading the teams, output the initial list of teams (ordered by team name).
4. While reading the week data, echo print the input. Also, print the winner, loser, winner score, and loser score.
5. After each week, show the current standings ordered by team name
6. Larry provided:
   * **p5Input.txt** - input file containing the data as described above.
7. Make certain your code works on a fox server.
8. Turn in a zip file named LastnameFirstname.zip containing:
   * **p5driver.py** - main driver which reads the input file and calls the other functions
   * **p5team.py** - contains functions for getting and printing the team data
   * **p5week.py** - contains functions for getting the weekly data and any other weekly functions
   * **p5out.txt** - contains your output

**Coding Requirements**

Your program must be separated into **multiple functions and source files**:

* The main driver must be in **p5driver.py**
* The code for getting team data and printing team data must be in **p5team.py.**
* The code for getting the weekly data and supporting functions must be in **p5week.py**.
* To **import code from another file**, use a Python statement like this:

from *fileName* import *funcName1, funcName2*

Note the actual file would be named *fileName.*py. Inotherwords, the filename specified on the **from** would be without the**.py**.

Please make certain you document your code. I am not providing a programming standard for Python; instead, we are curious what you think should be done.

Your output should be similar to the following sample (partial) output:

Initial

Team Wins Losses

MARS 0 0

MISS 0 0

NT 0 0

RICE 0 0

UTEP 0 0

UTSA 0 0

>>> WEEK 1

>>> RUNNERS UPSET RICE, 10 TO 3

UTSA RICE 10 3

>>> GREEN SLAUGHTERED BY EAGLES, 6 TO 48

MISS NT 48 6

>>> MINERS DEFEAT MARTIANS, 12 TO 7

UTEP MARS 12 7

>>> ENDWEEK

Week 1

Team Wins Losses

MARS 0 1

MISS 1 0

NT 0 1

RICE 0 1

UTEP 1 0

UTSA 1 0

>>> WEEK 2

>>> UTSA TOPS MINERS, 14 TO 13

UTSA UTEP 14 13

>>> EAGLES SOAR OVER OWLS, 21 TO 14

MISS RICE 21 14

>>> MARS LOSES TO MEAN GREEN, 21 - 13

NT MARS 21 13

>>> ENDWEEK

Week 2

Team Wins Losses

MARS 0 2

MISS 2 0

NT 1 1

RICE 0 2

UTEP 1 1

UTSA 2 0

>>> WEEK 3

>>> UTEP BEAT BY EAGLES, 23 TO 24

MISS UTEP 24 23

>>> UTSA SHUTS OUT HERD, 54 TO 0

UTSA MARS 54 0

>>> OWLS SLAUGHTERED BY MEAN GREEN, 36 TO 0

NT RICE 36 0

>>> ENDWEEK

Week 3

Team Wins Losses

MARS 0 3

MISS 3 0

NT 2 1

RICE 0 3

UTEP 1 2

UTSA 3 0