CSE 231 **Spring 2020**

Computer Project #6

This assignment focuses on the design, implementation and testing of a Python program that uses lists and tuples.

It is worth 45 points (4.5% of course grade) and must be completed no later than **11:59 PM on Monday, February 24, 2020.).**

Assignment Overview

In this assignment, you will practice with lists and tuples.

Assignment Background

Dr. Enbody is a hockey player and fan so this project is about hockey statistics using data from the National Hockey League (nhl.com). In hockey one important statistic is the number of points that a player gets in a game; points are the sum of the goals scored and the number of goals that the player assisted in scoring (called an "assist" and means that the player passed to another player who scored). In hockey total points is almost as important as total goals scored. This data set has the top players with respect to points-per-game over their career, but only considering players who played at least 100 games. The data is **not** sorted by points-per-game.

Assignment Specifications

You will develop a Python program that extracts and displays statistics from a data file. The basic concept is that you will read *all* the data into a list of lists and then extract data of interest from that master_list. We provide a file which is in the comma-separated-value format (file extension is `.csv`). You can open the file to look at it using a spreadsheet program such as Excel.

`open_file () → file pointer:`

a. This function takes no arguments, prompts for a file name and returns the file pointer to that file. The function will keep asking until a file is found. Use `try-except` and `FileNotFoundError`.
b. Parameters: none
c. Returns : file pointer
d. The function displays a prompt and possibly an error message.

`read_file (fp) → list of lists:`

a. This function accepts a file pointer and returns a master list of lists of all the data in the file. Each line of the file will be a list in the master list. The master list will be in the same order that the data is in the file. The file is a comma-separated-value file (`.csv`). However, some data fields have commas so you must use the `csv` module:

```
import csv   # place this line at the top of the program file

#place these lines in this function
reader = csv.reader(fp)
for line_list in reader:  # note that using csv.reader you get a list!
```

b. Parameters: file pointer
c. Returns : `master_list (list of lists)`
d. The function displays nothing.

shoots_left_right(master_list)→ (int,int):

a. This function accepts as input the master list and returns two integers representing the count of players who shoot left and those who shoot right. Return the data in that order. This data is found in the column labeled “`S/C`” in the file.
Algorithm: read each list in the master_list, extract the “`S/C`” data from index 1, and count whether the player shoots left or right.
b. Parameters: `master_list (list of lists)`
c. Returns : `int, int`
d. The function displays nothing.

position(master_list)---→ (int,int,int,int)

a. This function accepts as input the master list and returns four integers representing the count of players who play positions left-wing (“`L`”), right-wing (“`R`”), center (“`C`”), and defense (“`D`”). Return the data in that order. This data is found in the column labeled “`Pos`” in the file. The algorithm is similar to the previous function.
b. Parameters: `master_list (list of lists)`
c. Returns : `int, int, int, int`
d. The function displays nothing.

off_side_shooter(master_list)---→ (int,int)

a. This function accepts as input the master list and returns two integers representing the count of players who play the left-wing position (“`L`”) but shoot right (“`R`”) and those who play the right-wing position (“`R`”) but shoot left (“`L`”). Return the data in that order. This data is found in the column labeled “`S/C`” for shooting data and “`Pos`” for the position data in the file. The algorithm is similar to the previous function.
b. Parameters: `master_list (list of lists)`
c. Returns : `int, int`
d. The function displays nothing.

points_per_game(master_list)---→ list of tuples

a. This function accepts as input the master list and returns a sorted list of tuples where each tuple is of the form:
`(points-per-game, player name, position)`
The list will be **sorted** on points-per-game (highest to lowest) and only the **top ten** are returned, i.e. the list contains ten tuples in decreasing, sorted order. The player name is in the first column, i.e. labeled “`Player`” whereas the points-per-game is in the column

labeled "`P/GP`" (which stands for Points/Games_Played). *You must convert points-per-game to a float* before sorting. Python sorts on the first item in the tuple (which is what you want in this case). Remember to sort in decreasing order. You can use slicing to extract the first (top) ten from the sorted list. (Optional: if you want a challenge, you can write this function in one line; list comprehension is useful.)

b. Parameters: `master_list (list of lists)`
c. Returns : `list of tuples`
d. The function displays nothing.

games_played(master_list)---→ list of tuples

a. This function accepts as input the master list and returns a sorted list of tuples where each tuple is of the form:
`(games played, player name)`
The list will be **sorted** on games played (highest to lowest) and only the **top ten** are returned, i.e. the list contains ten tuples in decreasing, sorted order. The games played is in the column labeled "`GP`". The challenge of this function is that some players have played over a thousand games so commas appear in the values. *You must remove the comma and convert to an integer before sorting*. (Optional: if you want a further challenge, you can write this function in one line; list comprehension is useful.)
b. Parameters: `master_list (list of lists)`
c. Returns : `list of tuples`
d. The function displays nothing

shots_taken(master_list)---→ list of tuples

a. This function accepts as input the master list and returns a sorted list of tuples where each tuple is of the form:
`(shots taken, player name)`
The list will be **sorted** on shots taken (highest to lowest) and only the **top ten** are returned, i.e. the list contains ten tuples in decreasing, sorted order. The shots taken is in the column labeled "`S`". As in the previous function, some players have taken over a thousand shots so commas appear in the values so again you must remove the comma and convert to an integer before sorting. *However, the new challenge in this function is that some data is missing* (indicated by a pair of hyphens "`--`") so you must skip that data, i.e. ignore that player. In the earlier years of the NHL this data wasn't recorded. (Optional: if you want a further challenge, you can write this function in one line; list comprehension is useful.)
b. Parameters: `master_list (list of lists)`
c. Returns : `list of tuples`
d. The function displays nothing

main():

This function calls the above functions, first to open the file and read the data, and then to display the values. Data will be displayed in the order of the functions above. We provide a file named strings.txt that has the strings and the formatting for printing the data. Note that in order to print data in the thousands a comma is included in the format string, e.g.

```
"{:<20s}{:>16,d}"
```

Assignment Notes and Hints

1. The coding standard for CSE 231 is posted on the course website:

   http://www.cse.msu.edu/~cse231/General/coding.standard.html

Items 1-9 of the Coding Standard will be enforced for this project.

2. The program will produce reasonable and readable output, with appropriate labels for all values displayed.

3. We provide a `proj06.py` program for you to start with.

4. If you "hard code" answers, you will receive a grade of zero for the whole project. An example of hard coding is to simply print the approximate value of e rather than calculating it and then printing the calculated average.

Suggested Procedure

The last version of your solution is the program which will be graded by your TA.

You should use the ***Mimir*** *system to back up your partial solutions, especially if you are working close to the project deadline. That is the easiest way to ensure that you won't lose significant portions of your work if your machine fails or there are other last-minute problems.*

Assignment Deliverable

The deliverable for this assignment is the following file:

`proj06.py` – the source code for your Python program

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

Sample Output

Function Test read_file():
Input: "scoring_short.csv"
Output:

```
[['Adam Oates', 'R', 'C', '1,337', '341', '1,079', '664', '415', '1,420',
'2,392', '415', '79', '210', '10,512:35', '0.26', '0.81', '0.5', '0.31',
'1.06', '1.79', '0.31', '--', '--', '--'], ['Alex Ovechkin', 'R', 'L',
'1,129', '684', '569', '351', '218', '1,253', '5,444', '711', '2,972', '438',
'23,635:51', '0.61', '0.5', '0.31', '0.19', '1.11', '4.82', '0.63', '2.63',
'0.39', '20:56'], ['Alexander Mogilny', 'L', 'R', '990', '473', '559', '374',
'185', '1,032', '2,966', '432', '281', '68', '8,418:43', '0.48', '0.56',
'0.38', '0.19', '1.04', '3', '0.44', '--', '--', '--'], ['Artemi Panarin',
'R', 'L', '365', '140', '241', '151', '90', '381', '949', '114', '114', '72',
'7,150:45', '0.38', '0.66', '0.41', '0.25', '1.04', '2.6', '0.31', '0.31',
'0.2', '19:35'], ['Auston Matthews', 'L', 'C', '257', '142', '117', '75',
'42', '259', '886', '44', '89', '213', '4,742:55', '0.55', '0.46', '0.29',
'0.16', '1.01', '3.45', '0.17', '0.35', '0.83', '18:27'], ['Bernie Federko',
'L', 'C', '1,000', '369', '761', '469', '292', '1,130', '2,074', '487', '0',
'0', '--', '0.37', '0.76', '0.47', '0.29', '1.13', '2.07', '0.49', '--', '--',
'--'], ['Bernie Nicholls', 'R', 'C', '1,127', '475', '734', '458', '276',
'1,209', '3,231', '1,292', '65', '25', '1,011:55', '0.42', '0.65', '0.41',
'0.24', '1.07', '2.87', '1.15', '--', '--', '--'], ['Bill Cowley', 'L', 'C',
'549', '195', '354', '253', '99', '549', '--', '143', '0', '0', '--', '0.36',
'0.64', '0.46', '0.18', '1', '--', '0.26', '--', '--', '--'], ['Bobby Clarke',
'L', 'C', '1,144', '358', '852', '541', '311', '1,210', '2,582', '1,453', '0',
'0', '--', '0.31', '0.74', '0.47', '0.27', '1.06', '2.26', '1.27', '--', '--',
'--'], ['Bobby Hull', 'L', 'L', '1,063', '610', '560', '366', '194', '1,170',
'4,577', '634', '0', '0', '--', '0.57', '0.53', '0.34', '0.18', '1.1', '4.31',
'0.6', '--', '--', '--'], ['Bobby Orr', 'L', 'D', '657', '270', '645', '326',
'319', '915', '3,058', '953', '0', '0', '--', '0.41', '0.98', '0.5', '0.49',
'1.39', '4.65', '1.45', '--', '--', '--'], ['Brett Hull', 'R', 'R', '1,269',
'741', '650', '388', '262', '1,391', '4,876', '458', '142', '122', '9,648:19',
'0.58', '0.51', '0.31', '0.21', '1.1', '3.84', '0.36', '--', '--', '--'],
['Bryan Trottier', 'L', 'C', '1,279', '524', '901', '542', '359', '1,425',
'2,841', '912', '0', '0', '--', '0.41', '0.7', '0.42', '0.28', '1.11', '2.22',
'0.71', '--', '--', '--'], ['Connor McDavid', 'L', 'C', '333', '152', '290',
'184', '106', '442', '1,017', '110', '142', '133', '7,149:33', '0.46', '0.87',
'0.55', '0.32', '1.33', '3.05', '0.33', '0.43', '0.4', '21:28']]
```

Function Test shoots_left_right():

Input:

```
[['Adam Oates', 'R', 'C', '1,337', '341', '1,079', '664', '415', '1,420',
'2,392', '415', '79', '210', '10,512:35', '0.26', '0.81', '0.5', '0.31',
'1.06', '1.79', '0.31', '--', '--', '--'], ['Alex Ovechkin', 'R', 'L',
'1,129', '684', '569', '351', '218', '1,253', '5,444', '711', '2,972', '438',
'23,635:51', '0.61', '0.5', '0.31', '0.19', '1.11', '4.82', '0.63', '2.63',
'0.39', '20:56'], ['Alexander Mogilny', 'L', 'R', '990', '473', '559', '374',
'185', '1,032', '2,966', '432', '281', '68', '8,418:43', '0.48', '0.56',
'0.38', '0.19', '1.04', '3', '0.44', '--', '--', '--'], ['Artemi Panarin',
'R', 'L', '365', '140', '241', '151', '90', '381', '949', '114', '114', '72',
'7,150:45', '0.38', '0.66', '0.41', '0.25', '1.04', '2.6', '0.31', '0.31',
'0.2', '19:35'], ['Auston Matthews', 'L', 'C', '257', '142', '117', '75',
'42', '259', '886', '44', '89', '213', '4,742:55', '0.55', '0.46', '0.29',
'0.16', '1.01', '3.45', '0.17', '0.35', '0.83', '18:27'], ['Bernie Federko',
'L', 'C', '1,000', '369', '761', '469', '292', '1,130', '2,074', '487', '0',
'0', '--', '0.37', '0.76', '0.47', '0.29', '1.13', '2.07', '0.49', '--', '--',
'--'], ['Bernie Nicholls', 'R', 'C', '1,127', '475', '734', '458', '276',
'1,209', '3,231', '1,292', '65', '25', '1,011:55', '0.42', '0.65', '0.41',
'0.24', '1.07', '2.87', '1.15', '--', '--', '--'], ['Bill Cowley', 'L', 'C',
'549', '195', '354', '253', '99', '549', '--', '143', '0', '0', '--', '0.36',
'0.64', '0.46', '0.18', '1', '--', '0.26', '--', '--', '--'], ['Bobby Clarke',
'L', 'C', '1,144', '358', '852', '541', '311', '1,210', '2,582', '1,453', '0',
'0', '--', '0.31', '0.74', '0.47', '0.27', '1.06', '2.26', '1.27', '--', '--',
'--'], ['Bobby Hull', 'L', 'L', '1,063', '610', '560', '366', '194', '1,170',
'4,577', '634', '0', '0', '--', '0.57', '0.53', '0.34', '0.18', '1.1', '4.31',
'0.6', '--', '--', '--'], ['Bobby Orr', 'L', 'D', '657', '270', '645', '326',
'319', '915', '3,058', '953', '0', '0', '--', '0.41', '0.98', '0.5', '0.49',
'1.39', '4.65', '1.45', '--', '--', '--'], ['Brett Hull', 'R', 'R', '1,269',
'741', '650', '388', '262', '1,391', '4,876', '458', '142', '122', '9,648:19',
'0.58', '0.51', '0.31', '0.21', '1.1', '3.84', '0.36', '--', '--', '--'],
['Bryan Trottier', 'L', 'C', '1,279', '524', '901', '542', '359', '1,425',
'2,841', '912', '0', '0', '--', '0.41', '0.7', '0.42', '0.28', '1.11', '2.22',
'0.71', '--', '--', '--'], ['Connor McDavid', 'L', 'C', '333', '152', '290',
'184', '106', '442', '1,017', '110', '142', '133', '7,149:33', '0.46', '0.87',
'0.55', '0.32', '1.33', '3.05', '0.33', '0.43', '0.4', '21:28']]
```

Output:

```
(9, 5)
```

Function Test position():

Input:

```
[['Adam Oates', 'R', 'C', '1,337', '341', '1,079', '664', '415', '1,420',
'2,392', '415', '79', '210', '10,512:35', '0.26', '0.81', '0.5', '0.31',
'1.06', '1.79', '0.31', '--', '--', '--'], ['Alex Ovechkin', 'R', 'L',
'1,129', '684', '569', '351', '218', '1,253', '5,444', '711', '2,972', '438',
'23,635:51', '0.61', '0.5', '0.31', '0.19', '1.11', '4.82', '0.63', '2.63',
'0.39', '20:56'], ['Alexander Mogilny', 'L', 'R', '990', '473', '559', '374',
'185', '1,032', '2,966', '432', '281', '68', '8,418:43', '0.48', '0.56',
'0.38', '0.19', '1.04', '3', '0.44', '--', '--', '--'], ['Artemi Panarin',
'R', 'L', '365', '140', '241', '151', '90', '381', '949', '114', '114', '72',
'7,150:45', '0.38', '0.66', '0.41', '0.25', '1.04', '2.6', '0.31', '0.31',
'0.2', '19:35'], ['Auston Matthews', 'L', 'C', '257', '142', '117', '75',
'42', '259', '886', '44', '89', '213', '4,742:55', '0.55', '0.46', '0.29',
'0.16', '1.01', '3.45', '0.17', '0.35', '0.83', '18:27'], ['Bernie Federko',
'L', 'C', '1,000', '369', '761', '469', '292', '1,130', '2,074', '487', '0',
'0', '--', '0.37', '0.76', '0.47', '0.29', '1.13', '2.07', '0.49', '--', '--',
'--'], ['Bernie Nicholls', 'R', 'C', '1,127', '475', '734', '458', '276',
'1,209', '3,231', '1,292', '65', '25', '1,011:55', '0.42', '0.65', '0.41',
'0.24', '1.07', '2.87', '1.15', '--', '--', '--'], ['Bill Cowley', 'L', 'C',
'549', '195', '354', '253', '99', '549', '--', '143', '0', '0', '--', '0.36',
'0.64', '0.46', '0.18', '1', '--', '0.26', '--', '--', '--'], ['Bobby Clarke',
'L', 'C', '1,144', '358', '852', '541', '311', '1,210', '2,582', '1,453', '0',
'0', '--', '0.31', '0.74', '0.47', '0.27', '1.06', '2.26', '1.27', '--', '--',
'--'], ['Bobby Hull', 'L', 'L', '1,063', '610', '560', '366', '194', '1,170',
'4,577', '634', '0', '0', '--', '0.57', '0.53', '0.34', '0.18', '1.1', '4.31',
'0.6', '--', '--', '--'], ['Bobby Orr', 'L', 'D', '657', '270', '645', '326',
'319', '915', '3,058', '953', '0', '0', '--', '0.41', '0.98', '0.5', '0.49',
'1.39', '4.65', '1.45', '--', '--', '--'], ['Brett Hull', 'R', 'R', '1,269',
'741', '650', '388', '262', '1,391', '4,876', '458', '142', '122', '9,648:19',
'0.58', '0.51', '0.31', '0.21', '1.1', '3.84', '0.36', '--', '--', '--'],
['Bryan Trottier', 'L', 'C', '1,279', '524', '901', '542', '359', '1,425',
'2,841', '912', '0', '0', '--', '0.41', '0.7', '0.42', '0.28', '1.11', '2.22',
'0.71', '--', '--', '--'], ['Connor McDavid', 'L', 'C', '333', '152', '290',
'184', '106', '442', '1,017', '110', '142', '133', '7,149:33', '0.46', '0.87',
'0.55', '0.32', '1.33', '3.05', '0.33', '0.43', '0.4', '21:28']]
```

Output:

```
(3, 2, 8, 1)
```

Function Test off_side_shooter():

Input:

```
[['Adam Oates', 'R', 'C', '1,337', '341', '1,079', '664', '415', '1,420',
'2,392', '415', '79', '210', '10,512:35', '0.26', '0.81', '0.5', '0.31',
'1.06', '1.79', '0.31', '--', '--', '--'], ['Alex Ovechkin', 'R', 'L',
'1,129', '684', '569', '351', '218', '1,253', '5,444', '711', '2,972', '438',
'23,635:51', '0.61', '0.5', '0.31', '0.19', '1.11', '4.82', '0.63', '2.63',
'0.39', '20:56'], ['Alexander Mogilny', 'L', 'R', '990', '473', '559', '374',
'185', '1,032', '2,966', '432', '281', '68', '8,418:43', '0.48', '0.56',
'0.38', '0.19', '1.04', '3', '0.44', '--', '--', '--'], ['Artemi Panarin',
'R', 'L', '365', '140', '241', '151', '90', '381', '949', '114', '114', '72',
'7,150:45', '0.38', '0.66', '0.41', '0.25', '1.04', '2.6', '0.31', '0.31',
'0.2', '19:35'], ['Auston Matthews', 'L', 'C', '257', '142', '117', '75',
'42', '259', '886', '44', '89', '213', '4,742:55', '0.55', '0.46', '0.29',
'0.16', '1.01', '3.45', '0.17', '0.35', '0.83', '18:27'], ['Bernie Federko',
'L', 'C', '1,000', '369', '761', '469', '292', '1,130', '2,074', '487', '0',
'0', '--', '0.37', '0.76', '0.47', '0.29', '1.13', '2.07', '0.49', '--', '--',
'--'], ['Bernie Nicholls', 'R', 'C', '1,127', '475', '734', '458', '276',
'1,209', '3,231', '1,292', '65', '25', '1,011:55', '0.42', '0.65', '0.41',
'0.24', '1.07', '2.87', '1.15', '--', '--', '--'], ['Bill Cowley', 'L', 'C',
'549', '195', '354', '253', '99', '549', '--', '143', '0', '0', '--', '0.36',
'0.64', '0.46', '0.18', '1', '--', '0.26', '--', '--', '--'], ['Bobby Clarke',
'L', 'C', '1,144', '358', '852', '541', '311', '1,210', '2,582', '1,453', '0',
'0', '--', '0.31', '0.74', '0.47', '0.27', '1.06', '2.26', '1.27', '--', '--',
'--'], ['Bobby Hull', 'L', 'L', '1,063', '610', '560', '366', '194', '1,170',
'4,577', '634', '0', '0', '--', '0.57', '0.53', '0.34', '0.18', '1.1', '4.31',
'0.6', '--', '--', '--'], ['Bobby Orr', 'L', 'D', '657', '270', '645', '326',
'319', '915', '3,058', '953', '0', '0', '--', '0.41', '0.98', '0.5', '0.49',
'1.39', '4.65', '1.45', '--', '--', '--'], ['Brett Hull', 'R', 'R', '1,269',
'741', '650', '388', '262', '1,391', '4,876', '458', '142', '122', '9,648:19',
'0.58', '0.51', '0.31', '0.21', '1.1', '3.84', '0.36', '--', '--', '--'],
['Bryan Trottier', 'L', 'C', '1,279', '524', '901', '542', '359', '1,425',
'2,841', '912', '0', '0', '--', '0.41', '0.7', '0.42', '0.28', '1.11', '2.22',
'0.71', '--', '--', '--'], ['Connor McDavid', 'L', 'C', '333', '152', '290',
'184', '106', '442', '1,017', '110', '142', '133', '7,149:33', '0.46', '0.87',
'0.55', '0.32', '1.33', '3.05', '0.33', '0.43', '0.4', '21:28']]
```

Output:

```
(2, 1)
```

Function Test points_per_game():

Input:

```
[['Adam Oates', 'R', 'C', '1,337', '341', '1,079', '664', '415', '1,420',
'2,392', '415', '79', '210', '10,512:35', '0.26', '0.81', '0.5', '0.31',
'1.06', '1.79', '0.31', '--', '--', '--'], ['Alex Ovechkin', 'R', 'L',
'1,129', '684', '569', '351', '218', '1,253', '5,444', '711', '2,972', '438',
'23,635:51', '0.61', '0.5', '0.31', '0.19', '1.11', '4.82', '0.63', '2.63',
'0.39', '20:56'], ['Alexander Mogilny', 'L', 'R', '990', '473', '559', '374',
'185', '1,032', '2,966', '432', '281', '68', '8,418:43', '0.48', '0.56',
'0.38', '0.19', '1.04', '3', '0.44', '--', '--', '--'], ['Artemi Panarin',
'R', 'L', '365', '140', '241', '151', '90', '381', '949', '114', '114', '72',
'7,150:45', '0.38', '0.66', '0.41', '0.25', '1.04', '2.6', '0.31', '0.31',
'0.2', '19:35'], ['Auston Matthews', 'L', 'C', '257', '142', '117', '75',
'42', '259', '886', '44', '89', '213', '4,742:55', '0.55', '0.46', '0.29',
'0.16', '1.01', '3.45', '0.17', '0.35', '0.83', '18:27'], ['Bernie Federko',
'L', 'C', '1,000', '369', '761', '469', '292', '1,130', '2,074', '487', '0',
'0', '--', '0.37', '0.76', '0.47', '0.29', '1.13', '2.07', '0.49', '--', '--',
'--'], ['Bernie Nicholls', 'R', 'C', '1,127', '475', '734', '458', '276',
'1,209', '3,231', '1,292', '65', '25', '1,011:55', '0.42', '0.65', '0.41',
'0.24', '1.07', '2.87', '1.15', '--', '--', '--'], ['Bill Cowley', 'L', 'C',
'549', '195', '354', '253', '99', '549', '--', '143', '0', '0', '--', '0.36',
'0.64', '0.46', '0.18', '1', '--', '0.26', '--', '--', '--'], ['Bobby Clarke',
'L', 'C', '1,144', '358', '852', '541', '311', '1,210', '2,582', '1,453', '0',
'0', '--', '0.31', '0.74', '0.47', '0.27', '1.06', '2.26', '1.27', '--', '--',
'--'], ['Bobby Hull', 'L', 'L', '1,063', '610', '560', '366', '194', '1,170',
'4,577', '634', '0', '0', '--', '0.57', '0.53', '0.34', '0.18', '1.1', '4.31',
'0.6', '--', '--', '--'], ['Bobby Orr', 'L', 'D', '657', '270', '645', '326',
'319', '915', '3,058', '953', '0', '0', '--', '0.41', '0.98', '0.5', '0.49',
'1.39', '4.65', '1.45', '--', '--', '--'], ['Brett Hull', 'R', 'R', '1,269',
'741', '650', '388', '262', '1,391', '4,876', '458', '142', '122', '9,648:19',
'0.58', '0.51', '0.31', '0.21', '1.1', '3.84', '0.36', '--', '--', '--'],
['Bryan Trottier', 'L', 'C', '1,279', '524', '901', '542', '359', '1,425',
'2,841', '912', '0', '0', '--', '0.41', '0.7', '0.42', '0.28', '1.11', '2.22',
'0.71', '--', '--', '--'], ['Connor McDavid', 'L', 'C', '333', '152', '290',
'184', '106', '442', '1,017', '110', '142', '133', '7,149:33', '0.46', '0.87',
'0.55', '0.32', '1.33', '3.05', '0.33', '0.43', '0.4', '21:28']]
```

Output:

```
[(1.39, 'Bobby Orr', 'D'), (1.33, 'Connor McDavid', 'C'), (1.13, 'Bernie
Federko', 'C'), (1.11, 'Bryan Trottier', 'C'), (1.11, 'Alex Ovechkin', 'L'),
(1.1, 'Brett Hull', 'R'), (1.1, 'Bobby Hull', 'L'), (1.07, 'Bernie Nicholls',
'C'), (1.06, 'Bobby Clarke', 'C'), (1.06, 'Adam Oates', 'C')]
```

Function Test games_played():

Input:

```
[['Adam Oates', 'R', 'C', '1,337', '341', '1,079', '664', '415', '1,420',
'2,392', '415', '79', '210', '10,512:35', '0.26', '0.81', '0.5', '0.31',
'1.06', '1.79', '0.31', '--', '--', '--'], ['Alex Ovechkin', 'R', 'L',
'1,129', '684', '569', '351', '218', '1,253', '5,444', '711', '2,972', '438',
'23,635:51', '0.61', '0.5', '0.31', '0.19', '1.11', '4.82', '0.63', '2.63',
'0.39', '20:56'], ['Alexander Mogilny', 'L', 'R', '990', '473', '559', '374',
'185', '1,032', '2,966', '432', '281', '68', '8,418:43', '0.48', '0.56',
'0.38', '0.19', '1.04', '3', '0.44', '--', '--', '--'], ['Artemi Panarin',
'R', 'L', '365', '140', '241', '151', '90', '381', '949', '114', '114', '72',
'7,150:45', '0.38', '0.66', '0.41', '0.25', '1.04', '2.6', '0.31', '0.31',
'0.2', '19:35'], ['Auston Matthews', 'L', 'C', '257', '142', '117', '75',
'42', '259', '886', '44', '89', '213', '4,742:55', '0.55', '0.46', '0.29',
'0.16', '1.01', '3.45', '0.17', '0.35', '0.83', '18:27'], ['Bernie Federko',
'L', 'C', '1,000', '369', '761', '469', '292', '1,130', '2,074', '487', '0',
'0', '--', '0.37', '0.76', '0.47', '0.29', '1.13', '2.07', '0.49', '--', '--',
'--'], ['Bernie Nicholls', 'R', 'C', '1,127', '475', '734', '458', '276',
'1,209', '3,231', '1,292', '65', '25', '1,011:55', '0.42', '0.65', '0.41',
'0.24', '1.07', '2.87', '1.15', '--', '--', '--'], ['Bill Cowley', 'L', 'C',
'549', '195', '354', '253', '99', '549', '--', '143', '0', '0', '--', '0.36',
'0.64', '0.46', '0.18', '1', '--', '0.26', '--', '--', '--'], ['Bobby Clarke',
'L', 'C', '1,144', '358', '852', '541', '311', '1,210', '2,582', '1,453', '0',
'0', '--', '0.31', '0.74', '0.47', '0.27', '1.06', '2.26', '1.27', '--', '--',
'--'], ['Bobby Hull', 'L', 'L', '1,063', '610', '560', '366', '194', '1,170',
'4,577', '634', '0', '0', '--', '0.57', '0.53', '0.34', '0.18', '1.1', '4.31',
'0.6', '--', '--', '--'], ['Bobby Orr', 'L', 'D', '657', '270', '645', '326',
'319', '915', '3,058', '953', '0', '0', '--', '0.41', '0.98', '0.5', '0.49',
'1.39', '4.65', '1.45', '--', '--', '--'], ['Brett Hull', 'R', 'R', '1,269',
'741', '650', '388', '262', '1,391', '4,876', '458', '142', '122', '9,648:19',
'0.58', '0.51', '0.31', '0.21', '1.1', '3.84', '0.36', '--', '--', '--'],
['Bryan Trottier', 'L', 'C', '1,279', '524', '901', '542', '359', '1,425',
'2,841', '912', '0', '0', '--', '0.41', '0.7', '0.42', '0.28', '1.11', '2.22',
'0.71', '--', '--', '--'], ['Connor McDavid', 'L', 'C', '333', '152', '290',
'184', '106', '442', '1,017', '110', '142', '133', '7,149:33', '0.46', '0.87',
'0.55', '0.32', '1.33', '3.05', '0.33', '0.43', '0.4', '21:28']]
```

Output:

```
[(1337, 'Adam Oates'), (1279, 'Bryan Trottier'), (1269, 'Brett Hull'), (1144,
'Bobby Clarke'), (1129, 'Alex Ovechkin'), (1127, 'Bernie Nicholls'), (1063,
'Bobby Hull'), (1000, 'Bernie Federko'), (990, 'Alexander Mogilny'), (657,
'Bobby Orr')]
```

Function Test shots_taken():

Input:

```
[['Adam Oates', 'R', 'C', '1,337', '341', '1,079', '664', '415', '1,420',
'2,392', '415', '79', '210', '10,512:35', '0.26', '0.81', '0.5', '0.31',
'1.06', '1.79', '0.31', '--', '--', '--'], ['Alex Ovechkin', 'R', 'L',
'1,129', '684', '569', '351', '218', '1,253', '5,444', '711', '2,972', '438',
'23,635:51', '0.61', '0.5', '0.31', '0.19', '1.11', '4.82', '0.63', '2.63',
'0.39', '20:56'], ['Alexander Mogilny', 'L', 'R', '990', '473', '559', '374',
'185', '1,032', '2,966', '432', '281', '68', '8,418:43', '0.48', '0.56',
'0.38', '0.19', '1.04', '3', '0.44', '--', '--', '--'], ['Artemi Panarin',
'R', 'L', '365', '140', '241', '151', '90', '381', '949', '114', '114', '72',
'7,150:45', '0.38', '0.66', '0.41', '0.25', '1.04', '2.6', '0.31', '0.31',
'0.2', '19:35'], ['Auston Matthews', 'L', 'C', '257', '142', '117', '75',
'42', '259', '886', '44', '89', '213', '4,742:55', '0.55', '0.46', '0.29',
'0.16', '1.01', '3.45', '0.17', '0.35', '0.83', '18:27'], ['Bernie Federko',
'L', 'C', '1,000', '369', '761', '469', '292', '1,130', '2,074', '487', '0',
'0', '--', '0.37', '0.76', '0.47', '0.29', '1.13', '2.07', '0.49', '--', '--',
'--'], ['Bernie Nicholls', 'R', 'C', '1,127', '475', '734', '458', '276',
'1,209', '3,231', '1,292', '65', '25', '1,011:55', '0.42', '0.65', '0.41',
'0.24', '1.07', '2.87', '1.15', '--', '--', '--'], ['Bill Cowley', 'L', 'C',
'549', '195', '354', '253', '99', '549', '--', '143', '0', '0', '--', '0.36',
'0.64', '0.46', '0.18', '1', '--', '0.26', '--', '--', '--'], ['Bobby Clarke',
'L', 'C', '1,144', '358', '852', '541', '311', '1,210', '2,582', '1,453', '0',
'0', '--', '0.31', '0.74', '0.47', '0.27', '1.06', '2.26', '1.27', '--', '--',
'--'], ['Bobby Hull', 'L', 'L', '1,063', '610', '560', '366', '194', '1,170',
'4,577', '634', '0', '0', '--', '0.57', '0.53', '0.34', '0.18', '1.1', '4.31',
'0.6', '--', '--', '--'], ['Bobby Orr', 'L', 'D', '657', '270', '645', '326',
'319', '915', '3,058', '953', '0', '0', '--', '0.41', '0.98', '0.5', '0.49',
'1.39', '4.65', '1.45', '--', '--', '--'], ['Brett Hull', 'R', 'R', '1,269',
'741', '650', '388', '262', '1,391', '4,876', '458', '142', '122', '9,648:19',
'0.58', '0.51', '0.31', '0.21', '1.1', '3.84', '0.36', '--', '--', '--'],
['Bryan Trottier', 'L', 'C', '1,279', '524', '901', '542', '359', '1,425',
'2,841', '912', '0', '0', '--', '0.41', '0.7', '0.42', '0.28', '1.11', '2.22',
'0.71', '--', '--', '--'], ['Connor McDavid', 'L', 'C', '333', '152', '290',
'184', '106', '442', '1,017', '110', '142', '133', '7,149:33', '0.46', '0.87',
'0.55', '0.32', '1.33', '3.05', '0.33', '0.43', '0.4', '21:28']]
```

Output:

```
[(5444, 'Alex Ovechkin'), (4876, 'Brett Hull'), (4577, 'Bobby Hull'), (3231,
'Bernie Nicholls'), (3058, 'Bobby Orr'), (2966, 'Alexander Mogilny'), (2841,
'Bryan Trottier'), (2582, 'Bobby Clarke'), (2392, 'Adam Oates'), (2074,
'Bernie Federko')]
```

Test 1:

```
Enter filename: Scoring_per_Game.csv

 Shooting
left:    39
right:   25

  Position
left:       7
right:     18
center:    36
defense:    3

    Off-side Shooter
left-wing shooting right:    2
right-wing shooting left:    8

      Top Ten Points-Per-Game
Player              Position Points Per Game
Wayne Gretzky              C            1.92
Mario Lemieux              C            1.88
Mike Bossy                 R            1.50
Joe Malone                 C            1.40
Bobby Orr                  D            1.39
Connor McDavid             C            1.33
Marcel Dionne              C            1.31
Sidney Crosby              C            1.28
Peter Stastny              C            1.27
Peter Forsberg             C            1.25

        Top Ten Games-Played
Player                  Games Played
Gordie Howe                    1,767
Mark Messier                   1,756
Jaromir Jagr                   1,733
Ron Francis                    1,731
Steve Yzerman                  1,514
Wayne Gretzky                  1,487
Teemu Selanne                  1,451
Paul Coffey                    1,409
Stan Mikita                    1,396
Joe Sakic                      1,378

        Top Ten Shots-Taken
Player                   Shots Taken
Jaromir Jagr                   5,637
Alex Ovechkin                  5,444
```

```
Marcel Dionne                  5,363
Phil Esposito                  5,166
Wayne Gretzky                  5,088
Brett Hull                     4,876
Joe Sakic                      4,621
Steve Yzerman                  4,602
Bobby Hull                     4,577
Teemu Selanne                  4,540
```

Grading Rubric

```
Computer Project #06                             Scoring Summary
General Requirements:
  ( 4 pts) Coding Standard 1-9
     (descriptive comments, function headers, mnemonic identifiers,
     format, etc...)

Implementation:
 ( 4 pts)  open_file function (no Mimir test)

 ( 4 pts)  read_file function

 ( 4 pts)  shoots_left_right function

 ( 4 pts)  position function

 ( 4 pts)  off_side_shooter function

 ( 4 pts)  points_per_game function

 ( 4 pts)  games_played function

 ( 4 pts)  shots_taken function

 ( 9 pts)  Test 1


Note: hard coding an answer earns zero points for the whole project
-10 points for not using main()
```