CMSY 156 - Lab 7
Lists
(50 points)

Objective: Store and process test scores using lists and functions

There are seven functions required for this lab. A list must be used to store and process all test scores. All output must be the same as in the sample screenshots provided below.

1. **Function: main** (no params, no return value):

   The main function will first call the get_scores function and then the display_menu function. The main function will then prompt the user for a menu choice. The menu choice must be evaluated using a decision control structure to determine which function to execute. Each function is associated with one menu choice. After each menu choice and function call, the menu should redisplay using a repetition control structure. The program must continue executing until the user selects the Quit menu option.

2. **Function: get_scores**
   a) No parameters
   b) Return the list of scores
   c) Ask the user for the number of test scores being entered
   d) Repeatedly prompt the user to input scores
      until the specified number of scores has been
      entered
      (from step b)
   e) Store the scores in a list

```
How many scores will you be entering: 5
Enter test score 1 (0 - 100): 75
Enter test score 2 (0 - 100): 80
Enter test score 3 (0 - 100): 83
Enter test score 4 (0 - 100): 94
Enter test score 5 (0 - 100): 78

1. Score Metrics (min/max/avg)
2. Mine Scores (low/high scores)
3. Update Score
4. Display Scores
5. Quit
Enter your choice:
```

3. **Function: display_menu**
   a. No parameters
   b. No return value
   c. The input statement should be in the main function directly after the call to the display menu function.

   Display a menu with the following menu options.

```
1. Process Scores (min/max/avg)
2. Mine Scores (low/high scores)
3. Update Score
4. Display Scores
5. Quit
Enter your choice:
```

4. **Function: score_metrics**
   a) Parameter: Scores list
   b) No return value
   c) Find and display the highest, and lowest test scores.
   d) Calculate and display the average test score.

```
1. Score Metrics (min/max/avg)
2. Mine Scores (low/high scores)
3. Update Score
4. Display Scores (reverse order)
5. Quit
Enter your choice: 1


Number of Scores: 5

High Score:       94.00
Low Score:        75.00
Average Score:    82.00
```

5. **Function: mine_scores**
   a) Parameter: scores list
   b) no return value
   c) Use list comprehensions to create two new lists and display them:
      i. one with scores greater than or equal to the average score
      ii. one with scores less than the average score
      iii. Each list should be sorted to display the scores in ascending order

   d) No return value

   Note: The new lists created in the function are not returned.

   Use two separate list comprehensions to generate two lists: high test scores and low test scores. The high and low test scores are based on the average test score. The average test score should be included in the high test score list.

```
1. Score Metrics (min/max/avg)
2. Mine Scores (low/high scores)
3. Update Score
4. Display Scores (reverse order)
5. Quit
Enter your choice: 2


Top Scores
 83.00
 94.00

Bottom Scores
 75.00
 78.00
 80.00
```

6. **Function: update_score**
   a) Parameter: scores list
   b) no return value
   c) Display the test scores numbered
   d) Prompt the user for the score to update
   e) Update the selected score (by index)
   f) Redisplay the test scores after the update.

   Note: This function must check for the following exceptions:

   - ValueError
   - NameError
   - IndexError
   - Catch all other exceptions with a general exception handler

```
1. Score Metrics (min/max/avg)
2. Mine Scores (low/high scores)
3. Update Score
4. Display Scores (reverse order)
5. Quit
Enter your choice: 3

Update test score
1. 75.00
2. 80.00
3. 83.00
4. 94.00
5. 78.00
Select score number to update: 3
Enter new score: 75
1. 75.00
2. 80.00
3. 75.00
4. 94.00
5. 78.00
```

7. **Function: display_scores**
   a) Parameter: scores list
   b) no return value
   c) Display the test scores in reverse order using a slice operation with negative indices.

   Note: Do not use the reverse function for this operation.

   A list with the following values [75, 80, 83, 94, 78] should display as:

```
1. Score Metrics (min/max/avg)
2. Mine Scores (low/high scores)
3. Update Score
4. Display Scores (reverse order)
5. Quit
Enter your choice: 4

Scores in reverse order:
          78.00
          94.00
          83.00
          80.00
          75.00
```