Justin Blain

Professor Luke Donoho

SDEV 300 7383

April 8, 2024

## Lab 4

### Testing

| **Purpose** | **Input** | **Expected Output** | **Actual Output** |
| --- | --- | --- | --- |
| Standard Execution | | | |
| Phone number checker | phone, 1234567890, qwe-rty-uiop,  123-456-7890 | Phone Number Prompt,  Format Error,  Phone Number Prompt,  Format Error,  Phone Number Prompt, Confirmation,  Main Menu |  |
| Zip code checker | zop,  zip,  123456789,  12345-6789 | Selection Error,  Zip Code Prompt,  Format Error,  Zip code Prompt,  Confirmation,  Main Menu |  |
| Matrix functionality | | | |
| Enter the Matrix | Matrix,  1, 2, 3, 4, 5, 6, 7, 8, 9  1.1, 2.2, 3.3, 4.4, 5.5, 6.6, 7.7, 8.8, 9.9 | Matrix 1 prompt,  First entry:  [[1. 2. 3.]  [4. 5. 6.]  [7. 8. 9.]]  Matrix 2 prompt,  Second entry:  [[1.1 2.2 3.3]  [4.4 5.5 6.6]  [7.7 8.8 9.9]]  Matrix Menu Prompt | Hello and welcome!  Please make a selection from the following  1. Check a "phone" number's format  2. Check a "zip" code's format  3. Enter the "matrix"  4. "Exit" the program  matrix  Please enter the nine values for matrix 1 one digit at a time  1  2  3  4  5  6  7  8  9  First entry:  [[1. 2. 3.]  [4. 5. 6.]  [7. 8. 9.]]  Please enter the nine values for matrix 2 one digit at a time  1.1  2.2  3.3  4.4  5.5  6.6  7.7  8.8  9.9  Second entry:  [[1.1 2.2 3.3]  [4.4 5.5 6.6]  [7.7 8.8 9.9]]  Let's work:  Would you like to:  1. "Add" the matrices  2. "Sub"tract the matrices  3. Perform "matrix mult"iplication  4. Perform element by 'element mult"iplication  5. "Overwrite" the currently entered matrices  5. Return to the "main" menu  6. "Exit" the program |
| Add | (after the above), add | Result,  Transpose,  Row Means,  Column Means,  Matrix Menu | add  The result is  [[ 2.1 4.2 6.3]  [ 8.4 10.5 12.6]  [14.7 16.8 18.9]]  The transpose is  [[ 2.1 8.4 14.7]  [ 4.2 10.5 16.8]  [ 6.3 12.6 18.9]]  The rows means are: [ 8.4 10.5 12.6]  The columns means are: [ 4.2 10.5 16.8]  None  Let's work:  Would you like to:  1. "Add" the matrices  2. "Sub"tract the matrices  3. Perform "matrix mult"iplication  4. Perform element by 'element mult"iplication  5. "Overwrite" the currently entered matrices  5. Return to the "main" menu  6. "Exit" the program |
| Subtract | (after the above), sub | Result,  Transpose,  Row Means,  Column Means,  Matrix Menu | sub  The result is  [[-0.1 -0.2 -0.3]  [-0.4 -0.5 -0.6]  [-0.7 -0.8 -0.9]]  The transpose is  [[-0.1 -0.4 -0.7]  [-0.2 -0.5 -0.8]  [-0.3 -0.6 -0.9]]  The rows means are: [-0.4 -0.5 -0.6]  The columns means are: [-0.2 -0.5 -0.8]  None  Let's work:  Would you like to:  1. "Add" the matrices  2. "Sub"tract the matrices  3. Perform "matrix mult"iplication  4. Perform element by 'element mult"iplication  5. "Overwrite" the currently entered matrices  5. Return to the "main" menu  6. "Exit" the program  6. "Exit" the program |
| Matrix Multiplication | (after the above), matrix mult | Result,  Transpose,  Row Means,  Column Means,  Matrix Menu | matrix mult  The result is  [[ 33. 39.6 46.2]  [ 72.6 89.1 105.6]  [112.2 138.6 165. ]]  The transpose is  [[ 33. 72.6 112.2]  [ 39.6 89.1 138.6]  [ 46.2 105.6 165. ]]  The rows means are: [ 72.6 89.1 105.6]  The columns means are: [ 39.6 89.1 138.6]  None  Let's work:  Would you like to:  1. "Add" the matrices  2. "Sub"tract the matrices  3. Perform "matrix mult"iplication  4. Perform element by 'element mult"iplication  5. "Overwrite" the currently entered matrices  5. Return to the "main" menu  6. "Exit" the program |
| Element Multiplication | (after the above), element mult | Result,  Transpose,  Row Means,  Column Means,  Matrix Menu | element mult  The result is  [[ 1.1 4.4 9.9]  [17.6 27.5 39.6]  [53.9 70.4 89.1]]  The transpose is  [[ 1.1 17.6 53.9]  [ 4.4 27.5 70.4]  [ 9.9 39.6 89.1]]  The rows means are: [24.2 34.1 46.2]  The columns means are: [ 5.13333333 28.23333333 71.13333333]  None  Let's work:  Would you like to:  1. "Add" the matrices  2. "Sub"tract the matrices  3. Perform "matrix mult"iplication  4. Perform element by 'element mult"iplication  5. "Overwrite" the currently entered matrices  5. Return to the "main" menu  6. "Exit" the program |
| Exits | | | |
| From within the Matrix | exit | So long, and thanks for all the fish! |  |
| From the main menu | exit | So long, and thanks for all the fish! |  |

### Pylint

#### Output

-------------------------------------------------------------------

Your code has been rated at 10.00/10 (previous run: 9.90/10, +0.10)

#### Discussion

* Now that I know that there's a workaround for everything I know that 10 out of 10 according to pylint is achievable. For an unused iterator variable, I switched to an underscore, and for an anomalous backslash in a string that will eventually be compiled into a regex expression I used a double backslash to make the code more legible.

### Password Security

While I wish my test had achieved more nuanced results to give a better range of insight (12 characters and less secure salt may have helped achieve this), it seems sequential numbers and words are less secure than more randomized methods. A mix of upper and lower case letters, numbers, and special strings between 8 and 16 chars and at least sha3 encryption seemed to achieve the highest level of security that this experiment rendered.