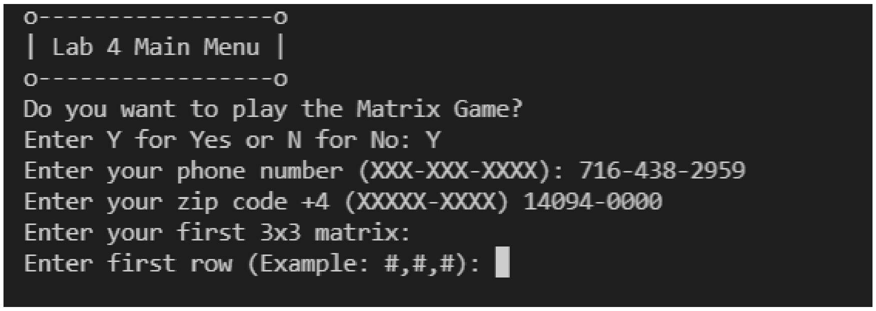
Allen Taylor
SDEV 300 6382 LAB 4
2/6/2022

Test Case:
Allows a user to enter and validate their phone number and zipcode+4.
Input:
Y, 716-438-2959, 14094-000
Expected Output:
User input is accepted, no errors thrown.
Actual Output:
User input is accepted, no errors thrown.



```
o-----------------o
| Lab 4 Main Menu |
o-----------------o
Do you want to play the Matrix Game?
Enter Y for Yes or N for No: Y
Enter your phone number (XXX-XXX-XXXX): 716-438-2959
Enter your zip code +4 (XXXXX-XXXX) 14094-0000
Enter your first 3x3 matrix:
Enter first row (Example: #,#,#): 
```

Test Case:

Input validation for phone number and zipcode+4.

Input:

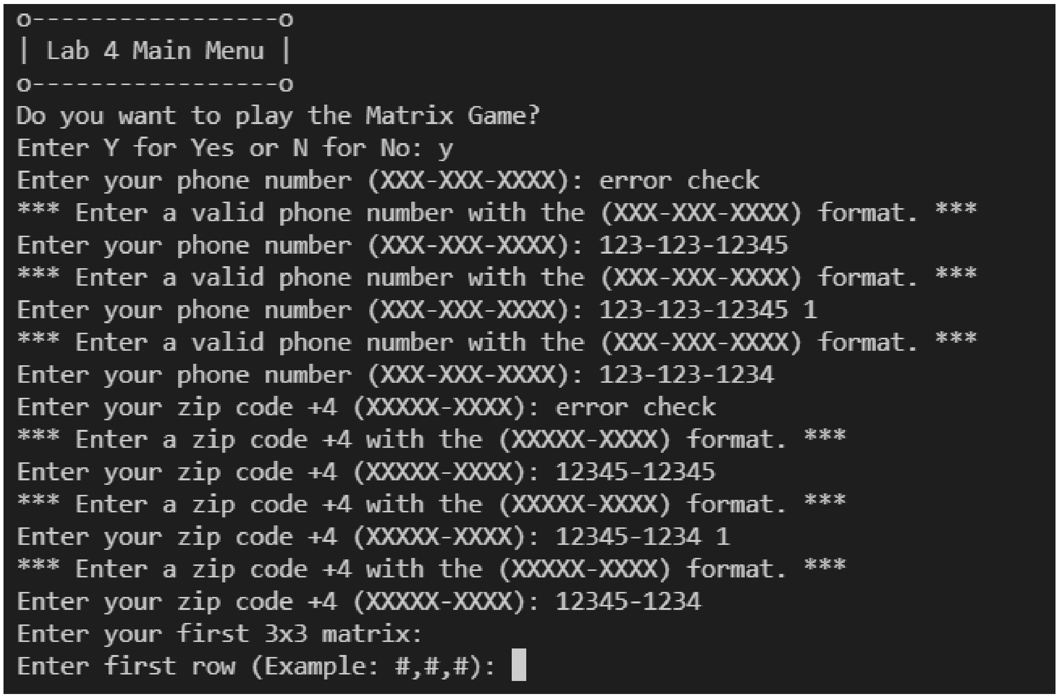
Y, error check, 123-123-12345, 123-123-12345 1, 123-123-1234, error check, 12345-12345, 12345-1234 1, 12345-1234

Expected Output:

Errors are caught.

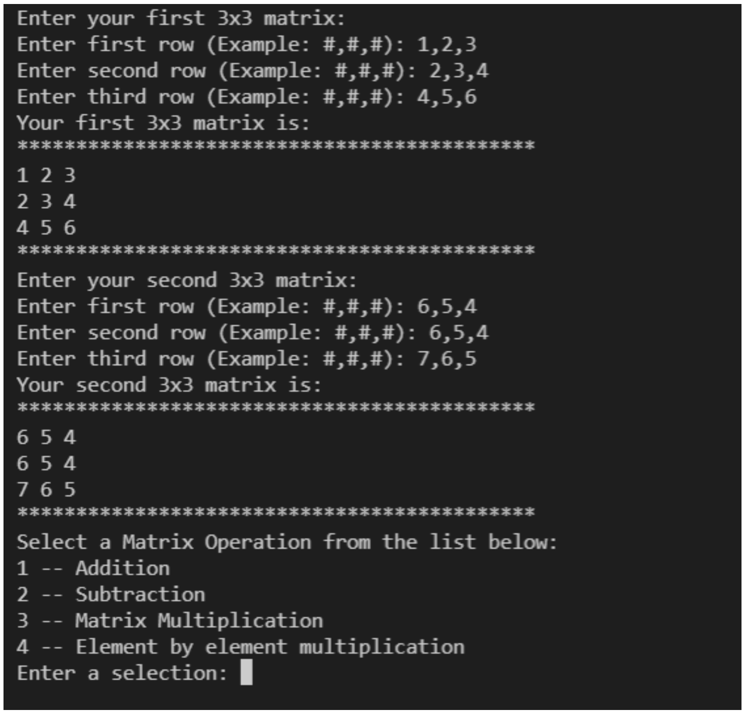
Actual Output:

Errors are caught.



```
o----------------o
| Lab 4 Main Menu |
o----------------o
Do you want to play the Matrix Game?
Enter Y for Yes or N for No: y
Enter your phone number (XXX-XXX-XXXX): error check
*** Enter a valid phone number with the (XXX-XXX-XXXX) format. ***
Enter your phone number (XXX-XXX-XXXX): 123-123-12345
*** Enter a valid phone number with the (XXX-XXX-XXXX) format. ***
Enter your phone number (XXX-XXX-XXXX): 123-123-12345 1
*** Enter a valid phone number with the (XXX-XXX-XXXX) format. ***
Enter your phone number (XXX-XXX-XXXX): 123-123-1234
Enter your zip code +4 (XXXXX-XXXX): error check
*** Enter a zip code +4 with the (XXXXX-XXXX) format. ***
Enter your zip code +4 (XXXXX-XXXX): 12345-12345
*** Enter a zip code +4 with the (XXXXX-XXXX) format. ***
Enter your zip code +4 (XXXXX-XXXX): 12345-1234 1
*** Enter a zip code +4 with the (XXXXX-XXXX) format. ***
Enter your zip code +4 (XXXXX-XXXX): 12345-1234
Enter your first 3x3 matrix:
Enter first row (Example: #,#,#):
```

Test Case:
User will enter values of two, 3x3 matrices and then select from options including, addition, subtraction, matrix multiplication, and element by element multiplication.
Input:
1,2,3
2,3,4
4,5,6
6,5,4
6,5,4
7,6,5
Expected Output:
State data and image are displayed.
Actual Output:
State data and image are displayed.



```
Enter your first 3x3 matrix:
Enter first row (Example: #,#,#): 1,2,3
Enter second row (Example: #,#,#): 2,3,4
Enter third row (Example: #,#,#): 4,5,6
Your first 3x3 matrix is:
********************************************
1 2 3
2 3 4
4 5 6
********************************************
Enter your second 3x3 matrix:
Enter first row (Example: #,#,#): 6,5,4
Enter second row (Example: #,#,#): 6,5,4
Enter third row (Example: #,#,#): 7,6,5
Your second 3x3 matrix is:
********************************************
6 5 4
6 5 4
7 6 5
********************************************
Select a Matrix Operation from the list below:
1 -- Addition
2 -- Subtraction
3 -- Matrix Multiplication
4 -- Element by element multiplication
Enter a selection: 
```

Test Case:

Input validation for 3x3 matrices.

Input:

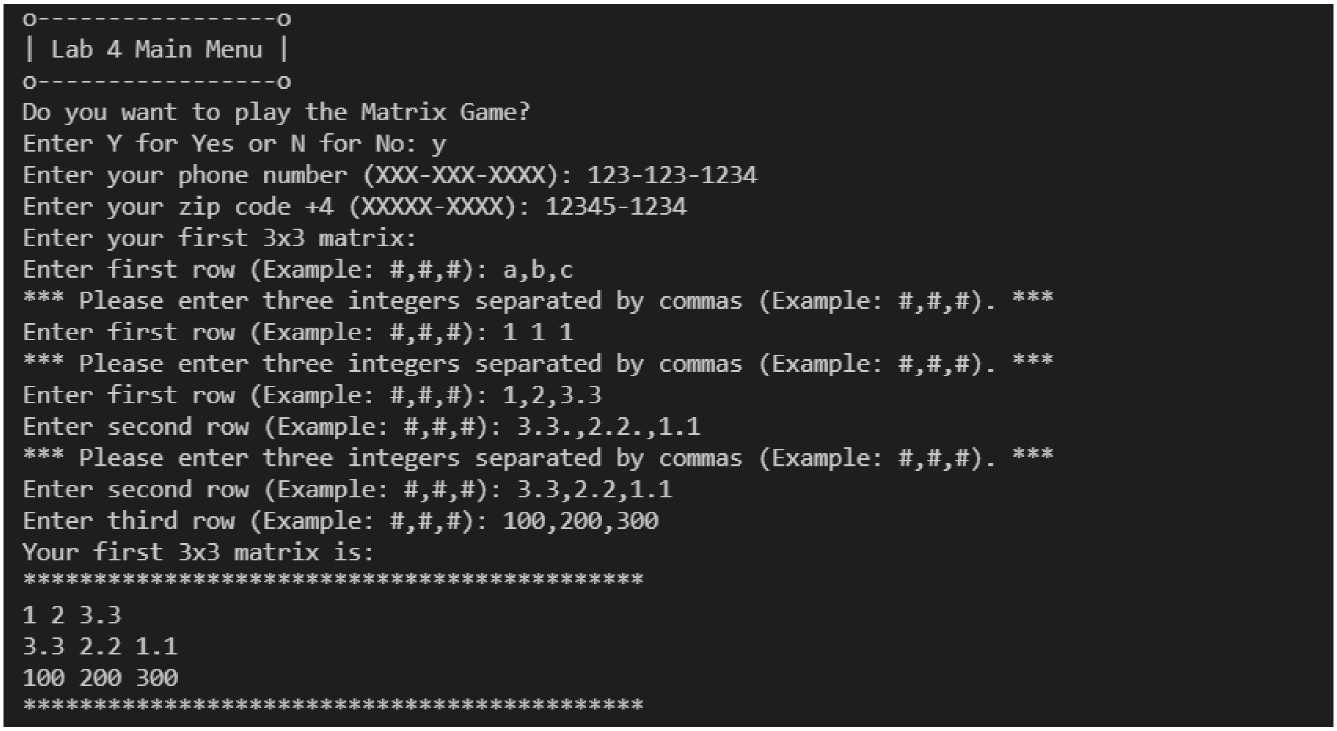
Y, 123-123-1234, 12345-1234
a,b,c
1 1 1
1,2,3.3
3.3.,2.2.,1.1
3.3,2.2,1.1
100,200,300

Expected Output:

Invalid inputs are caught.

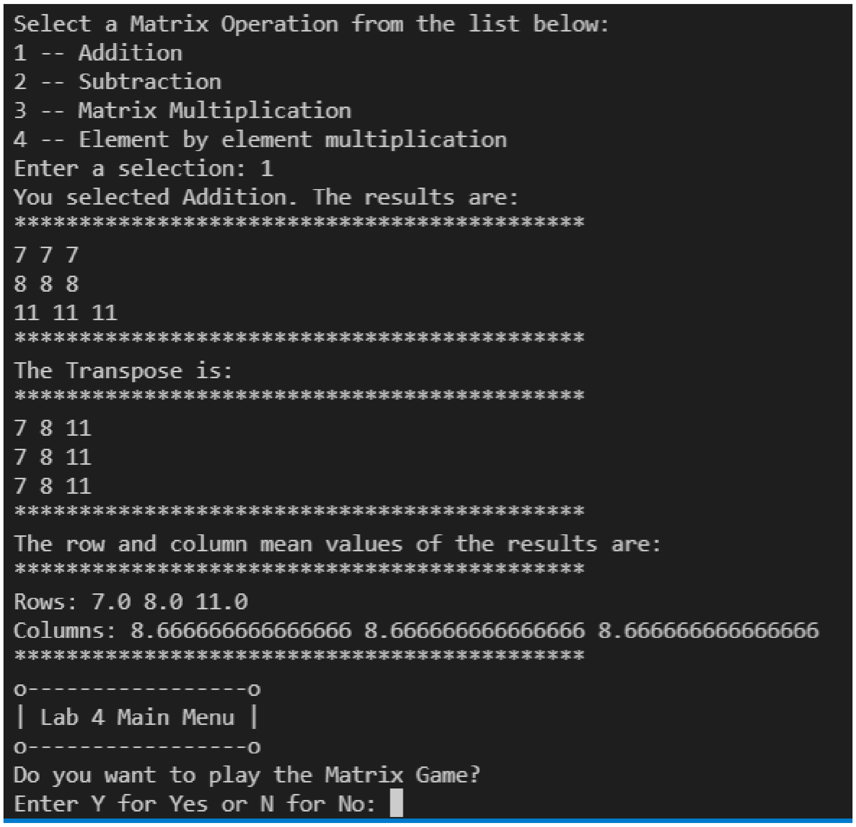
Actual Output:

Invalid inputs are caught.



```
o-----------------o
| Lab 4 Main Menu |
o-----------------o
Do you want to play the Matrix Game?
Enter Y for Yes or N for No: y
Enter your phone number (XXX-XXX-XXXX): 123-123-1234
Enter your zip code +4 (XXXXX-XXXX): 12345-1234
Enter your first 3x3 matrix:
Enter first row (Example: #,#,#): a,b,c
*** Please enter three integers separated by commas (Example: #,#,#). ***
Enter first row (Example: #,#,#): 1 1 1
*** Please enter three integers separated by commas (Example: #,#,#). ***
Enter first row (Example: #,#,#): 1,2,3.3
Enter second row (Example: #,#,#): 3.3.,2.2.,1.1
*** Please enter three integers separated by commas (Example: #,#,#). ***
Enter second row (Example: #,#,#): 3.3,2.2,1.1
Enter third row (Example: #,#,#): 100,200,300
Your first 3x3 matrix is:
********************************************
1 2 3.3
3.3 2.2 1.1
100 200 300
********************************************
```

Test Case:
-Compute and return the appropriate results for Addition.
-Return the transpose of the results.
-Return the mean of the rows for the results
-Return the mean of the columns for the results.
Input:
1
Expected Output:
Return valid results.
Actual Output:
Return valid results.



```
Select a Matrix Operation from the list below:
1 -- Addition
2 -- Subtraction
3 -- Matrix Multiplication
4 -- Element by element multiplication
Enter a selection: 1
You selected Addition. The results are:
********************************************
7 7 7
8 8 8
11 11 11
********************************************
The Transpose is:
********************************************
7 8 11
7 8 11
7 8 11
********************************************
The row and column mean values of the results are:
********************************************
Rows: 7.0 8.0 11.0
Columns: 8.666666666666666 8.666666666666666 8.666666666666666
********************************************
o-----------------o
| Lab 4 Main Menu |
o-----------------o
Do you want to play the Matrix Game?
Enter Y for Yes or N for No: 
```

Test Case:

-Compute and return the appropriate results for Subtraction.

-Return the transpose of the results.

-Return the mean of the rows for the results

-Return the mean of the columns for the results.

Input:

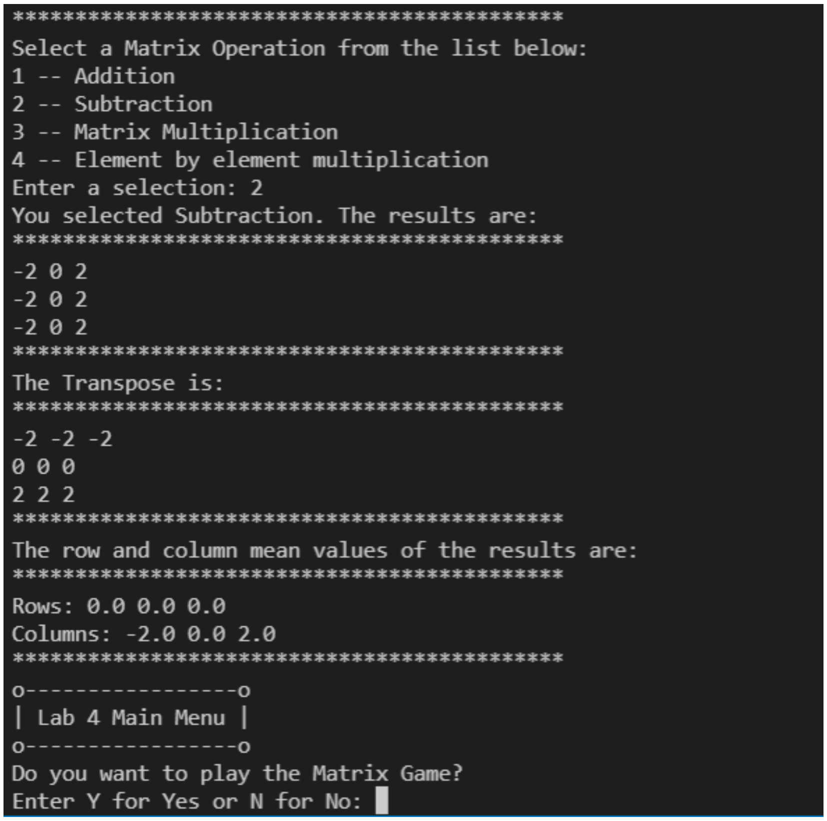
2

Expected Output:

Return valid results.

Actual Output:

Return valid results.



```
********************************************
Select a Matrix Operation from the list below:
1 -- Addition
2 -- Subtraction
3 -- Matrix Multiplication
4 -- Element by element multiplication
Enter a selection: 2
You selected Subtraction. The results are:
********************************************
-2 0 2
-2 0 2
-2 0 2
********************************************
The Transpose is:
********************************************
-2 -2 -2
0 0 0
2 2 2
********************************************
The row and column mean values of the results are:
********************************************
Rows: 0.0 0.0 0.0
Columns: -2.0 0.0 2.0
********************************************
o----------------o
| Lab 4 Main Menu |
o----------------o
Do you want to play the Matrix Game?
Enter Y for Yes or N for No: 
```

Test Case:

-Compute and return the appropriate results for Matrix Multiplication.

-Return the transpose of the results.

-Return the mean of the rows for the results

-Return the mean of the columns for the results.

Input:

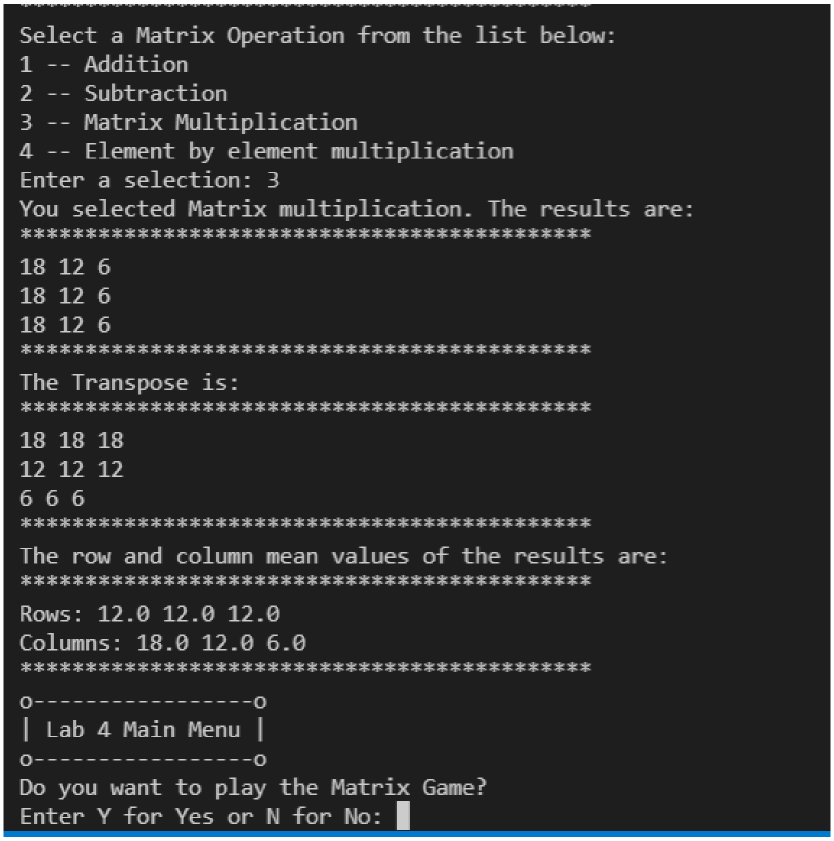
3

Expected Output:

Return valid results.

Actual Output:

Return valid results.



```
Select a Matrix Operation from the list below:
1 -- Addition
2 -- Subtraction
3 -- Matrix Multiplication
4 -- Element by element multiplication
Enter a selection: 3
You selected Matrix multiplication. The results are:
********************************************
18 12 6
18 12 6
18 12 6
********************************************
The Transpose is:
********************************************
18 18 18
12 12 12
6 6 6
********************************************
The row and column mean values of the results are:
********************************************
Rows: 12.0 12.0 12.0
Columns: 18.0 12.0 6.0
********************************************
o-----------------o
| Lab 4 Main Menu |
o-----------------o
Do you want to play the Matrix Game?
Enter Y for Yes or N for No:
```

Test Case:

-Compute and return the appropriate results for element by element Multiplication.

-Return the transpose of the results.

-Return the mean of the rows for the results

-Return the mean of the columns for the results.

Input:

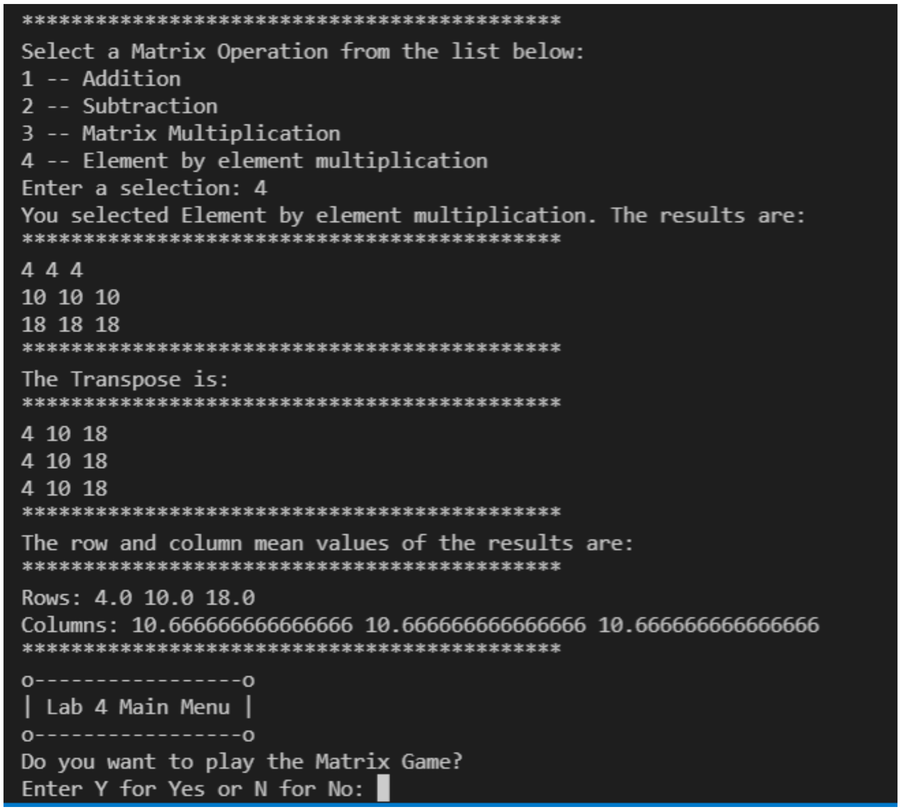
4

Expected Output:

Return valid results.

Actual Output:

Return valid results.



```
********************************************
Select a Matrix Operation from the list below:
1 -- Addition
2 -- Subtraction
3 -- Matrix Multiplication
4 -- Element by element multiplication
Enter a selection: 4
You selected Element by element multiplication. The results are:
********************************************
4 4 4
10 10 10
18 18 18
********************************************
The Transpose is:
********************************************
4 10 18
4 10 18
4 10 18
********************************************
The row and column mean values of the results are:
********************************************
Rows: 4.0 10.0 18.0
Columns: 10.666666666666666 10.666666666666666 10.666666666666666
********************************************
o-----------------o
| Lab 4 Main Menu |
o-----------------o
Do you want to play the Matrix Game?
Enter Y for Yes or N for No: 
```

Pylint Results: 9.72

```
************* Module lab_4
lab_4.py:226:0: C0301: Line too long (111/100) (line-too-long)
lab_4.py:214:0: R0912: Too many branches (16/12) (too-many-branches)

-----------------------------------------------------------------
Your code has been rated at 9.86/10 (previous run: 9.79/10, +0.07)

************* Module lab_4
lab_4.py:222:0: C0301: Line too long (111/100) (line-too-long)
lab_4.py:210:0: R0912: Too many branches (16/12) (too-many-branches)

-----------------------------------------------------------------
Your code has been rated at 9.86/10 (previous run: 9.86/10, +0.00)

************* Module lab_4
lab_4.py:241:15: C0303: Trailing whitespace (trailing-whitespace)
lab_4.py:242:18: C0303: Trailing whitespace (trailing-whitespace)
lab_4.py:256:0: C0301: Line too long (111/100) (line-too-long)
lab_4.py:238:11: E1124: Argument 'axis' passed by position and keyword in function call (redundant-keyword-arg)
lab_4.py:239:14: E1124: Argument 'axis' passed by position and keyword in function call (redundant-keyword-arg)
lab_4.py:244:0: R0912: Too many branches (18/12) (too-many-branches)
lab_4.py:244:0: R0915: Too many statements (51/50) (too-many-statements)

-----------------------------------------------------------------
Your code has been rated at 9.11/10 (previous run: 9.86/10, -0.76)

************* Module lab_4
lab_4.py:254:0: C0303: Trailing whitespace (trailing-whitespace)
lab_4.py:259:0: C0301: Line too long (115/100) (line-too-long)
lab_4.py:303:12: R1723: Unnecessary "elif" after "break" (no-else-break)
lab_4.py:248:0: R0912: Too many branches (16/12) (too-many-branches)
lab_4.py:248:0: R0915: Too many statements (56/50) (too-many-statements)

-----------------------------------------------------------------
```

Your code has been rated at 9.72/10 (previous run: 9.72/10, +0.00)

Password Cracking Activity Results

Password	Hash output	Did Crackstation work?
Password	5f4dcc3b5aa765d61d8327deb882cf99 5e884898da28047151d0e56f8dc6292773603d0d6aabbdd62a11ef721d1542d8 b109f3bbbc244eb82441917ed06d618b9008dd09b3befd1b5e07394c706a8bb9 80b1d7785e5976ec049b46df5f1326af5a2ea6d103fd07c95385ffab0cacbc86	YES
password123	482c811da5d5b4bc6d497ffa98491e38 ef92b778bafe771e89245b89ecbc08a44a4e166c06659911881f383d4473e94f bed4efa1d4fdbd954bd3705d6a2a78270ec9a52ecfbfb010c61862af5c76af176 1ffeb1aef6aca1bf5d02b3781aa854fabd2b69c790de74e17ecfec3cb6ac4bf	YES
Kocixatr	7c9f665474c8ab51fe39f6099280dde9 eef978ca5647fd29a68418cada0c8b3963487d6791901f78a6aa90be83502238 8acddddaee8220a2da9e73ac028d76bea04a20cf029e527ac1570badb7296ecd a565cfb8f77fdfa3a01536d8293d3e9f7e872baa0752bc02ab3b38d646711429	NO
siflgUql	dd3917a347fae72fe6bb667e6ac931b1 95fc732ee3d447f227f17c180f5c06829ebbe6124c4de5d508abbed4f861021c 3805e6296b5c308c74c4c99e170c289e101942c585f7bc674e4219a1e0d6abe f1b2f0bdd4c16bd6f35c493a12b940b68675b6de2f0214e5ae62aa0938744fd33	NO
f=oxujoC	2e47648543cd97a264fd1ee8bec4a836 16fb6d7654ed15a12d60ba6d89d4170845644b063d396a248be9403ead7236b3 6c387511067a6ba4356679bca7691a3679e6a748342b15daa376e361000b3963 d653bebcecf5bfd4022d2ce943a387824fc7ecf315dad9ba82c78a0c25f07fd5	NO
sw#s*a7E	8f36d293d5919b76392172b40307fafc f660b0348ea20066fe5dd83f8795bd34d4e92ad88f12affc238a9de4da3aa298 7213cb63d8e857452f92d6af0e03513c76f38206872e1f390a24d45b58971d2 39e83d8ae3805f40565b8ed575f49bbe9b16d8b799f6ea7546b9a6d9e63d9e33e	NO
P@55w0rd!	29ef52e7563626a96cea7f4b4085c124 98fbf2243b9d9a9f93b7d87dac20c8cdccb8315a0a5f02340a115f6a846113d0 7288e4efc6f8344eabb828a98ba1f5a46c25b3e398427aeee75e3a03c10c62b 6439c1dd14768020822d9ae1df9b327b0110369e0677e07d91c5e7cabc77adbba	YES
P@55w0rd123	35222e1b93b5bf885915f1774bb35dca 84e0d048216e0b6cdc89fcaa94c355744fae636508a44dcb1019a73a881b3476 2f3ff27b830197b24ce9048f22a57ab0492bec7ea2cda0ccf397ce9b226fd1c71a 0fae9a30451bb481b32a433decf5be7539f9f79269032f2d5be3cfe7bf89bb	YES
happysadglad	0ab1ead498a3e006ea9d7c4a7b2f74e4 c2d69d65724223c718e8c66b3873779eb728a373a7c2000a235b713e7d0f66bd a30d431acce765ea9322e3156bedea25128243aeea62ac84973ca32d2b971d07 f3ff56d6aa139e8a13ecac8ede290574684516bcbfb989b5ebfed0da329ce2e3	NO
supercalifragilisticexpialidocious	cbaf1a742147d7766e41dd48356b646a 91c459185b44be80e9604792d8d7c541c012e71bfb1e7c94627df24d5294a5f2 4c493b32f37b8adfdcbfa86eea0b3a13991e5ff80a959ed0182c1d4259f626738 5e292f9f562a697a02d0975c54348232a7c4e2cd517dc7e4ac824298c142f5f	NO
Mlh_Sw!19Wede**b	9265b9c6391f5ec0b477f64ab0b6b497 6a190295033cdbd9b105827b9389beb27c73ec64bf6bef2b4abdda658b57c99d c427888cca54478ca6948dc411fd338534c31603ebdfc5277facee7ee937151e34 040f7dd14636610ff88a51ef61049b6a8708a898509fc8c85468d80855d67f	NO
caK&v!18S3-fRO#H	d684192df9cc80a16b80d6ac7466a9ce eec33b3744eaed93823e78fc17d7fb43098dff7f135f64ac8e69274f41b2ac55 1550ee65ce66a58c5c07e4632815156a7ad7204484b08b3c23c9de2f149e9a bce2402223705e6c9a6521c50d744cca35bbaed67e02f6c84869f67d8d532f35dc	NO

stafu!u5oga=w*gO	a7b01500bd1fce59ce1da0fed7cb0157 646c5f388b91e2ffa566c5c572ef2955a3a2e02d72e916ba25de2ac810e24771 1e230c4c9466b86bd5f18fbcdd925471f2b38ba5b24939d0868b1a1ea348a9b cd773f3f166bd8880c9bcc4203aec98e3881982a7207c3539fedbd92f2d5c4c77	NO
rAweg1s9!lk=hi?r	fda824a8b9a7c7e2a8a7f3da4b38cf60 62e508495589856b8041db166c10897af1dc9464fcd3b637ed50afc1b3fca5b5 a0e8f4626ae94685006824c9f56aef015587b904d767c14192b22d53c3201ae adbcf36baf92873c03e9402f1cab8e71b0afe6b071cb92ff23836a51a1b988a54	NO
cerewRaju926a*um	4e45dea6a0b728087323588e6be848b1 624149c14b4a4f02d9c5e4c0ec6e876bf21a3297be8dd45a3400f6c1229b44e6 879cc7d473c08cd8dfaaeac526b97e2287695886a8d2882d4ebdf3c61a03076a db1dabf3e7a263bee632deb307626aa7e12dc094d05e681f0badeb93865b4561	NO
ch_fr4phEx$b+was	ff71cdf831fdd2d9b33151576c72f95d 62ae0614e17782bbc7e85077d926fd4b100c7978c981795500e1ed78747b4a0d 048271f1ef15f25c52845c63191f9f60d38a6a0d08e6d263d0be3b075d92b9cd1d 92aca7736f809d706d684b5c89f0d4642cf1b8f946e148426e55e709967449	NO
$IbrEW1e&?17pa1=	631a6f7ecd250c36b495909e6b9485e6 f6094cb3619bd94038163fab8f9bc62ceb5535235ce0158ebbf38b7e8298fccc ae582ce40c79fe98aed779bf779883a3928d5263162a1995678a6211ac0fc4a eca822597aed3f01b3b569de108f69ce51ec3ed81d7970dac91a26347504a60c3	NO
go1i09=3ph=hek_Q	1d213ee28ed10976908d32ca0ef179a9 9ae7d28542150eafeb771e1b6ae715a7c2edc7a0e270a23923320d14371da184 f375321c95588d63fafef7f1aa0c7c157d033df07d6e40bcf3a160afa7f47aae8803 de0e8d376cad03e7a6cdd891464a10aacadb3f932acfa3e5673bc149bac2	NO
br&6ixewrO#opr0w	d9852517032ed0a0e0621d14186dee20 87c501d087a6f367c4cdea0872ac2c684d09d5f2e3ceb6a61861270d161419e0 526643a0968867d1c5faddae356f55cd0b13d37a07aa4323683513b976187aa 840d2c6ab08ae229666f4599a13668ab17cfb08b9b8f0d8ffd184fbd9fac642e7	NO
Cr+cHUc4lp49*Ros	3a792e35072e2b3b8c56b51b25289ba2 abc04572b59458108b257feee3815d8a33d8d26ff9636cad88a344a062d640de 759a8730107d9897a70565c6734924af1731402003ac9d545a6611f7fa87020b 5ee6d8f6c2006a2d1326468717c0a6715b3e0d56398de3fe7c40776535c0fece	NO

Be sure to describe what you learned from this password cracking activity and what would you recommend as possible strong passwords after completing this activity in your report?

From my experiment with the hash cracker, any password that had “password” in it (in any form) could be cracked.

None of the passwords generated with Norton’s password generator had matching hashes.

General guidelines for strong passwords are:

- 12+ characters long
- Uses uppercase and lowercase letters, numbers and special symbols.
- Passwords that consist of mixed characters are harder to crack.
- Doesn't contain memorable keyboard paths.
- Is not based on personal information.