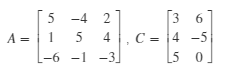
**Problem 1**

Download the live script file [MAT343lab1ex1Part1.mlx](https://math.la.asu.edu/~mat343/LAB1/MAT343lab1ex1Part1.mlx)  and open it with matlab. Run the file and use the information provided to complete the following exercise.

Enter the matrices



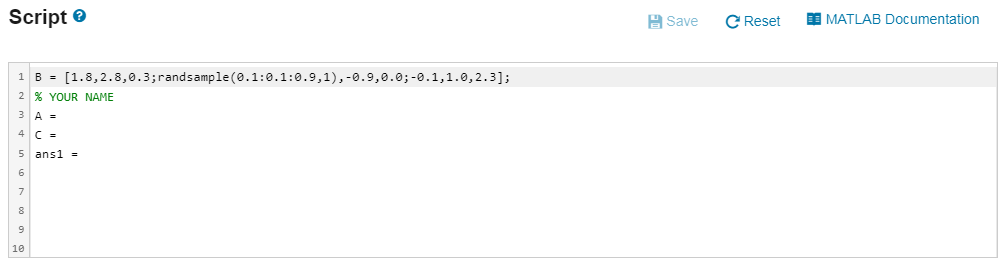
A matrix *B* is already entered for you (note that the matrix is ).

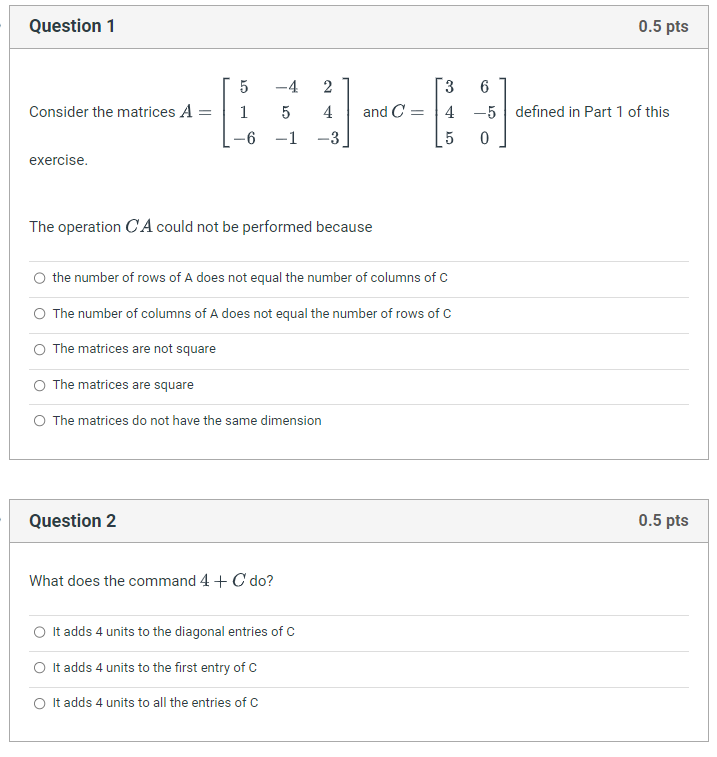
Of the following list of operations, **compute only those that make sense dimension-wise**, i.e. only those for which MATLAB would not give an error message. For each operation that make sense, store the results in variables called ans1, ans2, ... for questions 1 and 2, respectively. For instance, ans1 = B+A.

If an operation does not make sense dimension-wise delete or comment out that particular answer. For instance, if ans3 does not make sense, there should be no ans3 in your code.

Make sure you write down your results as you will be asked specific questions about them in the second part of this problem.

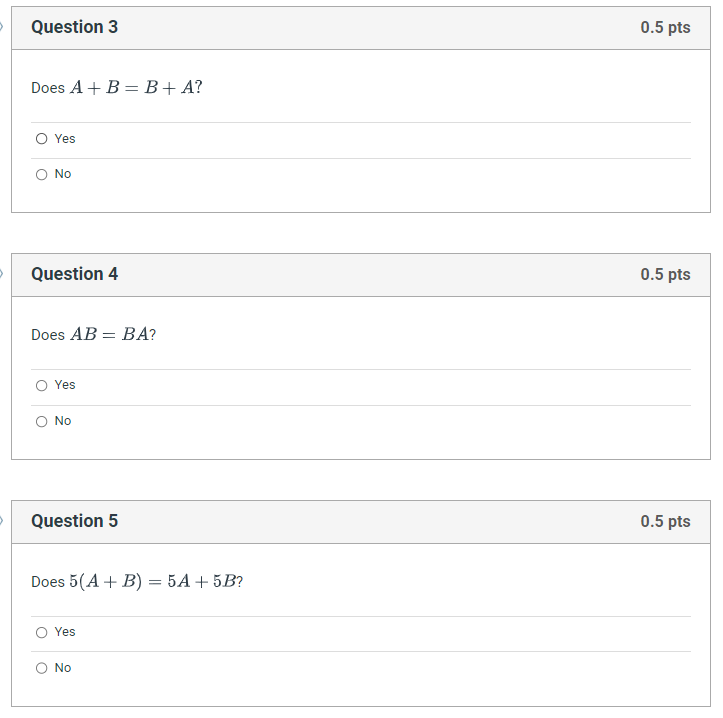
1. B+A
2. AB
3. BA
4. 4+C
5. A+C
6. CA
7. AC
8. 5(A+B)
9. A+B
10. 5A+5B





Question 1 답 The Number of rows of A does not equal the number of columns of C

Question 2 답 It adds 4 units to all the entries of C



Question 3 Yes

Question 4 No

Question 5 Yes

## Problem 2

The following rules of algebra 1-5 listed below hold for real numbers. However, most of them are false for matrices. Use the matrices  below to perform the appropriate operations in MATLAB and determine which rules are false. Below, 0 denotes the  zero matrix. Note that if a rule holds for these particular matrices, it does not mean that it is true in general. However, you should be able to determine that from the theory you have learned in class.

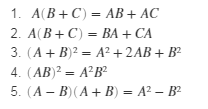
Enter the matrices ,

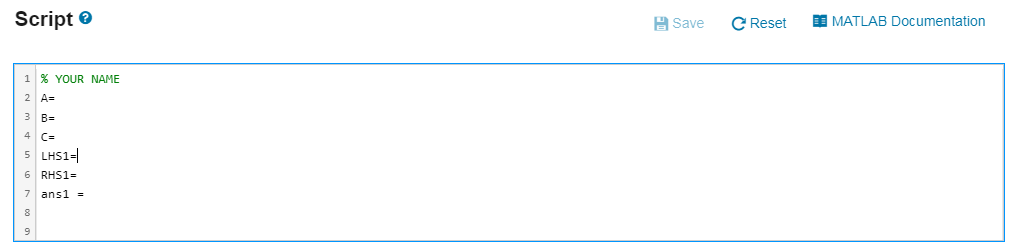


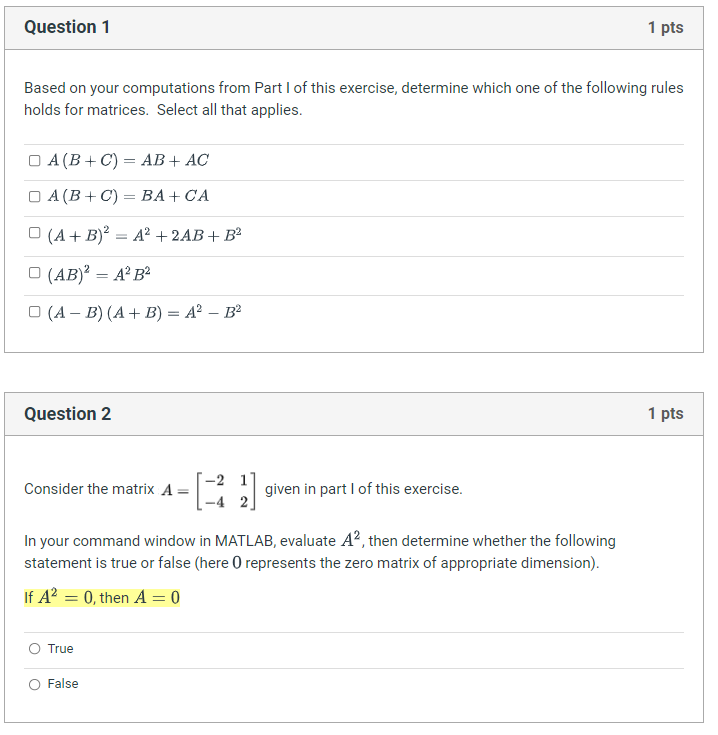
For each question below

* Compute the Left Hand Side and the Right Hand Side of the equation and store them in the variables LHS1, RHS1, LHS2, RHS2 and so on
* Use the logical operator == to determine whether the statements below are true or false. LHS==RHS  does element by element comparison between the variables LHS and RHS and returns an array with elements set to 1 (True) where the relation is true and elements set to 0 (False) where it is not. Store the result of question 1 in the variable ans1, the result of question 2 in ans2, and so on. (e.g. ans1=LHS1==RHS1).

You will be asked specific written questions about your computations in part II of this problem.

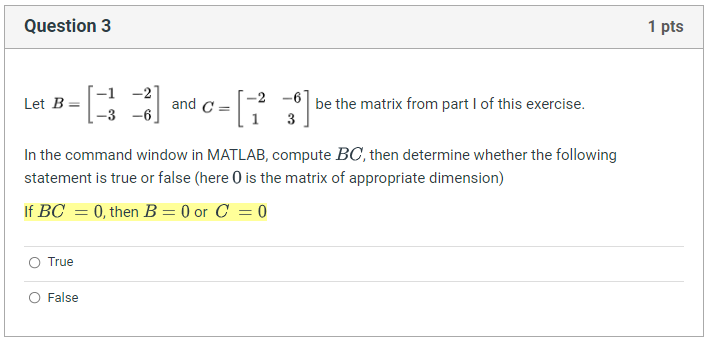






Question 1. A(B+C) = AB + AC

Question 2. False



Question 3. False

## Problem 3

The transpose of a matrix *A*, denoted by AT  , can be computed in MATLAB using A’

A matrix *A* is symmetric if AT = A. In this exercise we will explore properties of the transposition.

Enter the following matrices:

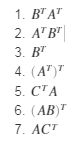


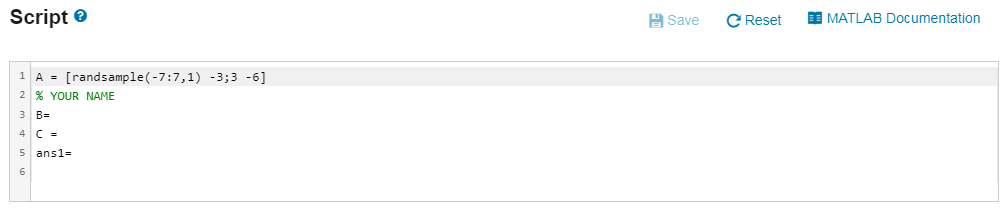
A matrix *A* is already entered for you (note that the matrix is ).

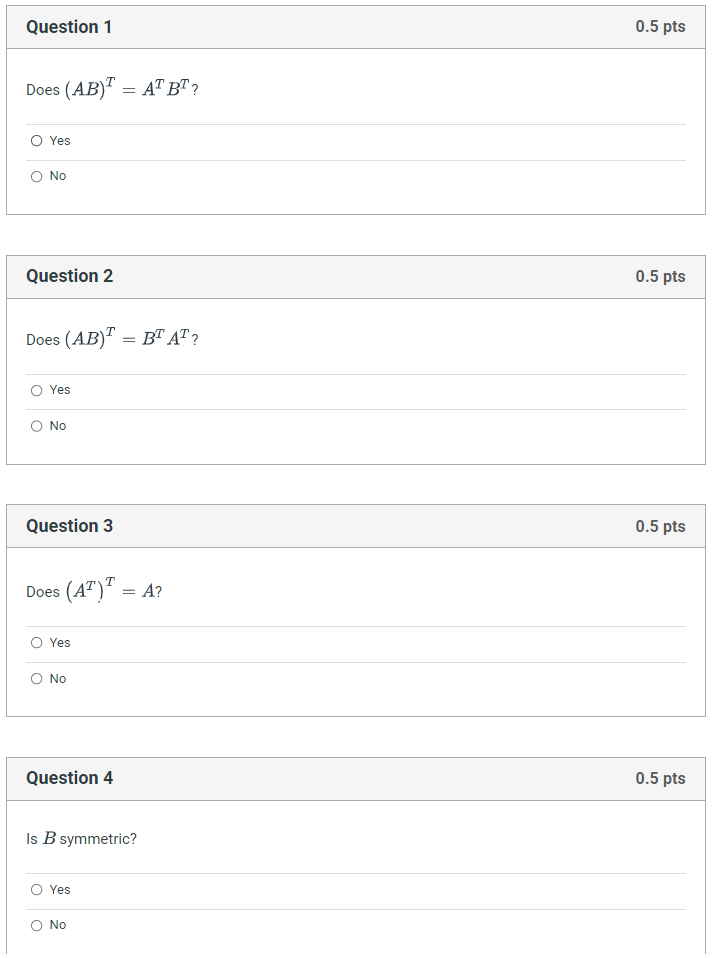
Of the following list of operations, compute only those that make sense dimension-wise, i.e. only those for which MATLAB would not give an error message. For each operation that make sense, store the results in variables called ans1, ans2, ...for questions 1 and 2 respectively. For instance, ans1 = B+A

If an operation does not make sense, do not include the answer for that question in the code.

Make sure you write down your results as you will be asked specific questions in Part II of this exercise. Also, **make sure you use prime ' to compute the transpose**.







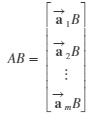
Q1 No, Q2 Yes, Q3 Yes, Q4 Yes

**Problem 4**

Download the live script file [MAT343lab1ex4.mlx](https://math.la.asu.edu/~mat343/LAB1/MAT343lab1ex4.mlx)

 and open it with matlab. Run the file and use the information provided to complete the following exercise.

Given two matrices *A* and *B*, let *C* be the product C=AB  (here we assume the dimensions of *A* and *B* are such that the multiplication is defined). The product AB can be computed in two different ways:

1. Multiplication by columns: Let bj and cj  be the jth columns of the matrices *B* and *C* respectively. By definition of matrix-matrix multiplication, we have cj = A bj, i.e. each column of the matrix C is given by the product of the matrix A times the corresponding colum of the matrix B: AB = [Ab1 Ab2 ….. Abn ]
2. Multiplication by rows: Let and be the i-th rows of the matrices *A* and *C* respectively. By definition of matrix-matrix multiplication, we have ,  i.e. each row of the matrix *C* is given by the product of corresponding row of *A* times the matrix *B*: 

In this exercise we will explore the two methods of computing the product.

Two 3 × 3 random matrices with entries between 0 and 10 are generated for you using the commands:

R =10\*rand(3), S=10\*rand(3)

The command rand(3) generates a random 3 × 3 matrix with entries in between 0 and 1. We multiply that matrix by 10 so that it has entries between 0 and 10.

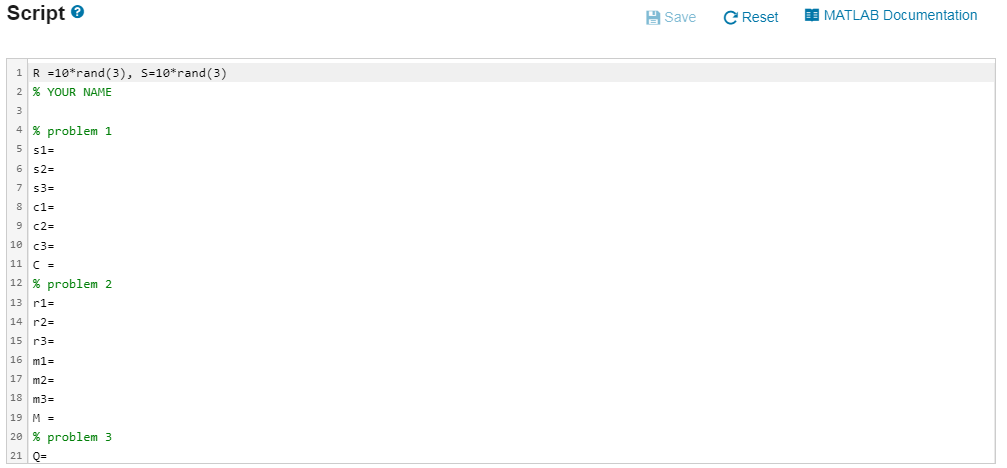
1. Multiplication by columns:

* Use the : operator to extract the first, second and third column of the matrix S and store them in the variables s1, s2 and s3 respectively (do not try to enter the single entries of the columns individually).
* Compute the products R\*s1, R\*s2, R\*s3 and store them in the varibles c1, c2, c3 respectively.
* Build the matrix C with columns c1, c2 and c3 (Note: build the matrix by concatenating the columns, do not enter the single entries individually)

2. Multiplication by rows:

* Use the : operator to extract the first, second and third rows of the matrix R and store them in the variables r1, r2 and r3 respectively (do not try to enter the single entries of the rows individually)
* Compute the products r1\*S, r2\*S, r3\*S and store them in the variables m1, m2, m3 respectively.
* Build the matrix M with rows m1, m2 and m3 (Note: build the matrix by concatenating the rows, do not enter the single entries individually)

3. Compute the product Q=RS and visually compare it with the matrices C and M.



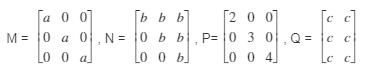
Q 와 C 와 M 모두 동일한 결과 입니다.

## Problem 5

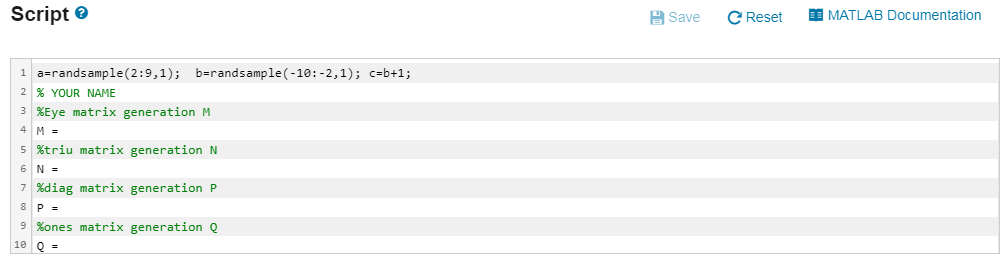
Download the live script file [MAT343lab1ex5.mlx](https://math.la.asu.edu/~mat343/LAB1/MAT343lab1ex5.mlx)  and open it with matlab. Run the file and use the information provided to complete the following exercise.

Create the following matrices with the help of the matrix generation functions **eye , ones, diag and triu**. **See the on-line help on these functions if required** (i.e. help eye). Do not enter the matrices explicitly. Each matrix should be generated with a single command.

M = , N = , P=, Q =



The variables  and *c* are already created for you.



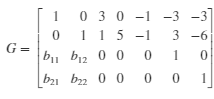
**Problem 6**

Download the live script file [MAT343lab1ex6.mlx](https://math.la.asu.edu/~mat343/LAB1/MAT343lab1ex6.mlx)  and open it with matlab. Run the file and use the information provided to complete the following exercise.

Enter the matrices .

A 2 X 2 matrix *B* has already been create for you.

The following matrix *G* is created by inserting the matrices *A*, *B*, and *C*, together with zero matrices and identity matrices in the appropriate position. Create the matrix using submatrices A, B, C, zeros and eye (that is, you are not allowed to enter the numbers explicitly). The matrix G should be created with a single command.





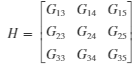
G = vertcat(horzcat(eye(2), C, A), horzcat(B, zeros(size(C)), eye(2)))

## Problem 7

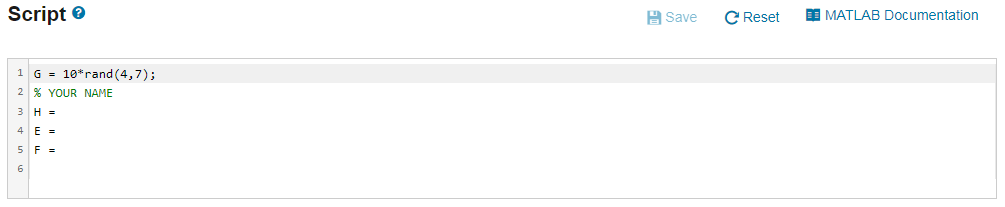
Download the live script file [MAT343lab1ex7.mlx](https://math.la.asu.edu/~mat343/LAB1/MAT343lab1ex7.mlx)  and open it with matlab. Run the file and use the information provided to complete the following exercise.

A  4 X 7 matrix G has been created for you.

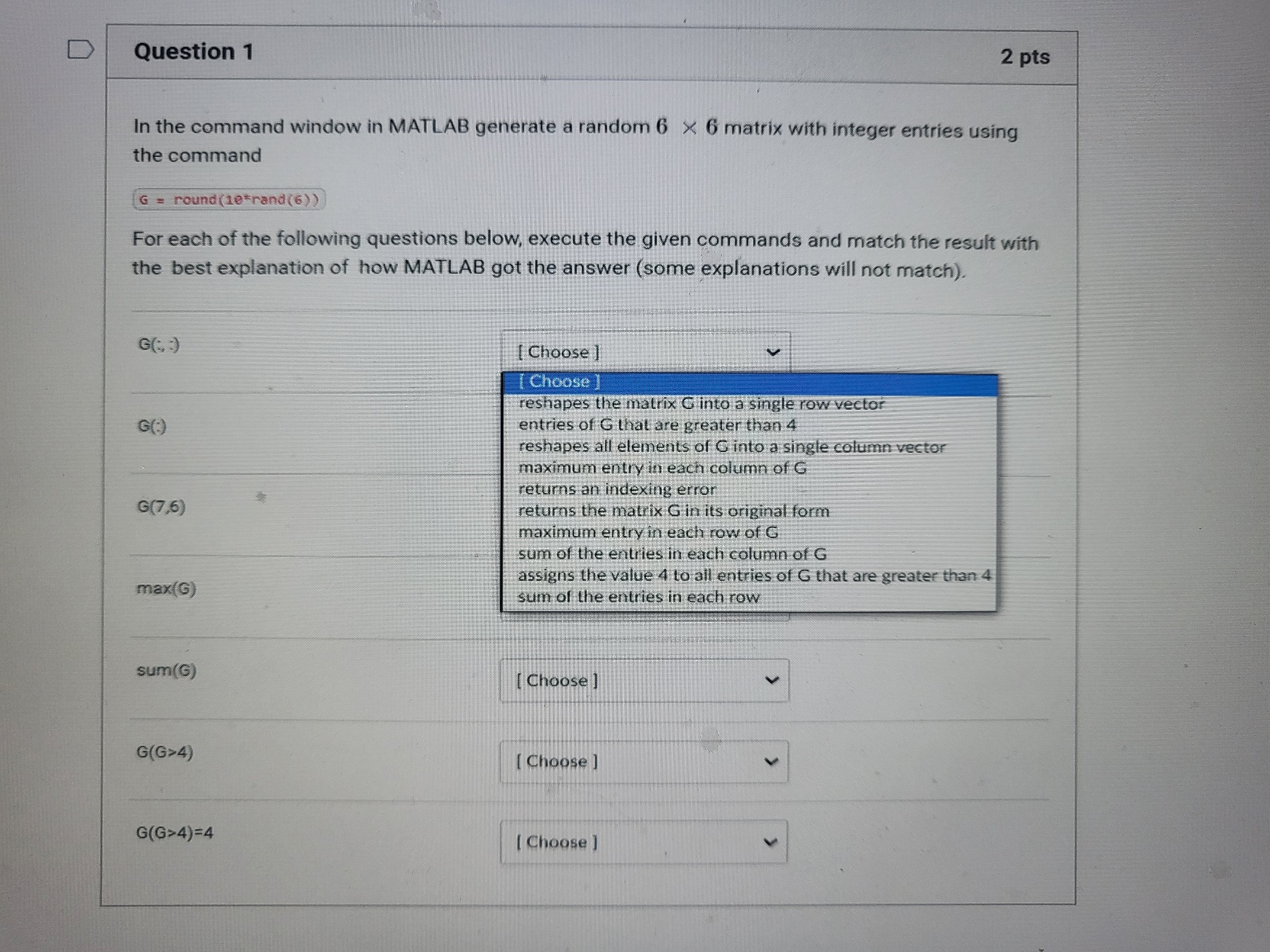
1. Extract the 3 X 3 submatrix of *G* consisting of columns 3 through 5 and rows 1 through 3 and store it in the matrix *H*, that is, create a matrix

by extracting the appropriate rows and columns from the matrix G.

1. Create the matrix *E* obtained from *H* by replacing the current value of H13 with 100. Do not enter *E* explicitly. **Hint:** enter first E=H; to create a copy of the matrix H and then manipulate the matrix E.
2. Create the matrix F obtained by deleting the second column of the matrix H. Do not enter F explicitly. **Hint:** enter first F=H; to create a copy of the matrix H and then manipulate the matrix F. Make sure you delete the column and **not** extract a submatrix.



Download the live script file  [MAT343lab1ex7Part2.mlx](https://math.la.asu.edu/~mat343/LAB1/MAT343lab1ex7Part2.mlx) and open it with matlab. Run the file and use the information provided to complete this quiz.



ans1 = G(:, :);% return the matrix G in its original form

ans2 = G(:);% reshapes all elements of G into a single column vector

%ans3 = G(7,6);% returns an indexing error

ans4 = max(G); % maximum entry in each column of G

ans5 = sum(G); % sum of the entries in each column of G

ans6 = G(G>4); % entries of G that are greater than 4

G(G>4)=4; % assigns the value 4 to all entries of G that are greater than 4

## Problem 8

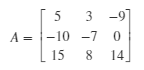
The three elementary row operations can be performed in MATLAB using the following commands

**Type I: A([i,j],:)=A([j,i],:)** interchanges row i and row j

**Type II: A(i,:)=α\*A(i,:)**multiplies row i by α

**Type III**: **A(i,:)=A(i,:)+α\*A(j,:)**  multiplies row j by α and adds it to row i

*The matrix A below has been created for you.*



Perform row operations in MATLAB that reduce the matrix A to **Row Echelon Form**.

The row operations are indicated for you in the learner template. Make sure you follow the instructions and perform only the row operation indicated in the comments. After each row operation, the result is stored in a new matrix in order for MATLAB to verify the accuracy of the intermediate steps.

You might want to click on "Run Script" at each row operations to check that your steps are correct and to determine the next step in the row reduction.

**NOTE:** in format rat, a number very close to zero, but not exactly zero due to round off error is denoted by \*. If you see \* in the output of your calculations, you should treat it as zero.

