Feedback — V. Octave Tutorial

Help

You submitted this quiz on **Fri 27 Jun 2014 7:23 PM PDT**. You got a score of **5.00** out of **5.00**.

Question 1

Suppose I first execute the following Octave commands:

```
A = [1 2; 3 4; 5 6];
B = [1 2 3; 4 5 6];
```

Which of the following are then valid Octave commands? Check all that apply. (Hint: `A'` denotes the transpose of A.)

Your Answer		Score	Explanation
☐ C = B + A;	✔	0.25	B is 2x3 and A is 3x2, so their sum is not well defined.
☑ C = A * B;	✔	0.25	A is 3x2 and B is 2x3, so A has the same number of columns as B has rows, and the product is well defined.
☑ C = B * A;	✔	0.25	B is 2x3 and A is 3x2, so B has the same number of columns as A has rows, and the product is well defined.
☐ C = A' * B;	✔	0.25	A' is 2x3 and B is 2x3, so A' does not have the same number of columns as B has rows, and the product is not well defined.
Total		1.00 / 1.00	

Question 2

Let $A = \begin{bmatrix} 16 & 2 & 3 & 13 \\ 5 & 11 & 10 & 8 \\ 9 & 7 & 6 & 12 \\ 4 & 14 & 15 & 1 \end{bmatrix}$.

Which of the following indexing expressions gives $B = \begin{bmatrix} 16 & 2 \\ 5 & 11 \\ 9 & 7 \\ 4 & 14 \end{bmatrix}$? Check all that apply.

Your Answer		Score	Explanation
☐ B = A(0:2, 0:4);	✔	0.25	The first element in Octave has index 1, so this expression is invalid.
☑ B = A(:, 1:2);	✔	0.25	A(:, 1:2) selects every row and the first two columns of A, giving the desired B.
☑ B = A(1:4, 1:2);	✔	0.25	A(1:4, 1:2) selects the first four rows and first two columns of A, giving the desired B.
☐ B = A(:, 0:2);	✔	0.25	The first element in Octave has index 1, so selecting columns 0 through 2 is invalid.
Total		1.00 / 1.00	

Question 3

Let A be a 10x10 matrix and x be a 10-element vector. Your friend wants to compute the product Ax and writes the following code:

```
v = zeros(10, 1);
for i = 1:10
  for j = 1:10
```

```
    v(i) = v(i) + A(i, j) * x(j);
  end
end
```

How would you vectorize this code to run without any `for` loops? Check all that apply.

Your Answer		Score	Explanation
v = A * x;	✔	0.25	Octave will correctly perform the matrix-vector product equivalent to the for loop above.
v = x' * A;	✔	0.25	This is a well-defined product to compute a 10-vector, but it computes a different set of values.
v = A .* x ;	✔	0.25	The .* operator performs element-wise multiplication, which is invalid for two matrices of different sizes.
v = Ax;	✔	0.25	Octave does not implicitly multiply without * but instead will look for a variable called "Ax".
Total		1.00 / 1.00	

Question 4

Say you have two column vectors v and w, each with 7 elements (i.e., they have dimensions 7x1). Consider the following code:

```
z = 0;
for i = 1:7
  z = z + v(i) * w(i);
end
```

Which of the following vectorizations correctly compute `z`? Check all that apply.

Your Answer		Score	Explanation
z = v * w	✔	0.25	v has dimension 7x1 and w has dimension 7x1, so their product is undefined.

;

☑ z = v' * w;	✔	0.25	By taking the transpose of v, the product computes the sum of the element-wise product of v and w, just as the for-loop code does.
☑ z = w' * v;	✔	0.25	By taking the transpose of w, the product computes the sum of the element-wise product of w and v, just as the for-loop code does.
☐ z = v * w ';	✔	0.25	v has dimension 7x1 and w' has dimension 1x7, so their product is a 7x7 matrix.
Total		1.00 / 1.00	

Question 5

In Octave, many functions work on single numbers, vectors, and matrices. For example, the `sin` function when applied to a matrix will return a new matrix with the sin of each element. But you have to be careful, as certain functions have different behavior. Suppose you have an 7x7 matrix X. You want to compute the log of every element, the square of every element, add 1 to every element, and divide every element by 4. You will store the results in four matrices, A, B, C, D. One way to do so is the following code:

```
for i = 1:7
  for j = 1:7
    A(i, j) = log (X(i, j));
    B(i, j) = X(i, j) ^ 2;
    C(i, j) = X(i, j) + 1;
    D(i, j) = X(i, j) / 4;
  end
end
```

Which of the following correctly compute A, B, C, or D? Check all that apply.

Your Answer		Score	Explanation
☑	✔	0.25	Adding a single number applies element-wise to a matrix.

C = X + 1;			
☐ B = X ^ 2;	✔	0.25	The code X ^ 2 is equivalent to X * X which is only defined if X is a square matrix. To compute the square of each element, you need to write X .^ 2.
☑ B = X .^ 2;	✔	0.25	The .^ operator perfoms element-wise exponentiation.
☑ A = log (X);	✔	0.25	The log function acts element-wise on matrix inputs.
Total		1.00 / 1.00	