PROFESSOR: Hi, everyone. Today, we're going to talk about linear transformations. So, we've seen linear transformations incognito all the time until now. We've played around with matrices. Matrices multiplying vectors, say, in R^n and producing vectors in R^m.

So really, the language of linear transformations only provides a nicer framework when we want to analyze linear operations on more abstract vector spaces, like the one we have in this problem here. We're going to work with the space of two by two matrices. And we're going to analyze the operation, have the matrix A, and we produce its transpose. OK. So please take a few minutes to try the problem on your own and come back.

Hi, again. OK. So the first question we need to ask ourselves is, indeed, why is T a linear operator? So what are the abstract properties that a linear operator satisfies? Well, what happens when T acts on the sum of two matrices, A and B?

So it produces the matrix the transpose of A plus B. But we know that this is A transpose, B transpose. And so, this is exactly T(A) plus T(B). So the transformation that we're analyzing takes the sum of two matrices into the sum of their transformations. OK. Similarly, it takes a multiple of a transformation into the multiple of the transformations. So it takes the matrix c times A to c times A transpose, which is c T of A.

OK. So it is a linear operator. Now, can we figure out what its inverse is? Well, what does the transpose do? It takes a column and flips it into a row. So what happens if we apply the operation once again? Well, it's going to take the row and turn it back down to the column.

So applying the transformation twice, we come back to the original situation. So therefore, T squared is the identity. And from this, we infer that the inverse is the transformation itself.

Now, this was part one. Part two, we'll compute the matrix of the linear transformation in the following two bases. So the first basis is, in fact-- it is the standard basis for the space of two by two matrices. And the way we compute the matrix, we first compute what T does to each of the basis elements. So T of v_1. Let's go back. So here.

So T takes the transpose of this matrix. And we see that the transpose of [1, 0; 0, 0] is [1, 0; 0, 0]. So it's a symmetric matrix. So T of v_1 is v_1. What about T of v_2? Come back here. So this 1 comes here. 0 comes here. And so we actually get v_3. So T of v_2 is v_3. Similarly, T

of v_3 is v_2.

And finally, T of v_4. Well, v_4 is a symmetric matrix as well. So the transpose doesn't change it. OK. Now, we encode this into a matrix in the following way. Essentially, the first column will tell us how T of v_1 is expressed as a linear combination of the basis elements.

Well, in this case, it's just v_1. So it's going to be 1 times v_1 plus 0*v_2 plus 0*v_3 plus 0*v_4. T of v_2 is v_3. So we have 0, 0, 1, 0. T of v_3 is 0*v_1, 1*v_2, 0*v_3, 0*v_4. And T of v4 is 0*v_1, 0*v_2, 0*v_3, plus 1*v_4.

OK. So we've written down the matrix of the linear transformation T in the standard basis. And you can check that this is exactly what we want. The representation of some matrix, say, [1, 2; 3, 4] in this standard basis is, it's the vector [1, 2, 3, 4]. T takes this to its transpose, [1, 3; 2, 4].

So this in the basis is represented as [1, 3, 2, 4]. Right? And it's not hard to see that when M_T multiplies this vector, we get exactly this vector. So we'll pause for a bit, so that I erase the board. And we're going to return with the representation of T in the basis w_1, w_2, w_3, and w_4.

OK. So let's now compute the matrix T in the basis w_1, w_2, w_3, and w_4. We played the same game. We look at how T acts on each of the basis vectors. So T of w_1-- well, w_1 is a symmetric matrix. So T of w_1 is w_1. Similarly, with w_2 and w_3. They're all symmetric.

What about w_4? Well, we see that the 1 comes down here, the negative one comes up here, and in the end, we just get the negative of w_4. So, let me just write this out. We had T of w_1 equal to w_1, T of w_2 equal to w_2, T of w_3 equal to w_3, and T of w_4, was negative of w_4.

So therefore, the matrix of the linear transformation T, in this basis-- I'm going to call the matrix M prime T-- has a fairly simple expression. The only non-zero entries are on a diagonal. And they're precisely 1, 1, 1, and negative 1.

And finally, let's tackle the eigenvalues slash eigenvectors issue. Well, you've seen what an eigenvector for a matrix is. And the idea for an eigenvalue, eigenvector for a linear transformation is virtually the same.

And we are looking for the vectors v and scalars lambda such that T of v is lambda*v. But if you guys look back to what we just did with w_1, w_2, w_3, and w_4, you'll see precisely that

w_1, w_2, and w_3 are eigenvectors for T with eigenvalue 1. And w_4 is an eigenvector for T with eigenvalue negative 1.

So yeah, we essentially have solved the problem knowing a very, very nice basis in which we computed the linear transformation T. So I'll leave it at that.