PROFESSOR: Let's continue our examination of GCD's and linear combinations and Euclidean algorithm by examining what's often called the extended Euclidean algorithm. It's a good name for it. Its ancient name, dating back to ancient India, is the pulverizer. And we will see what that does in a moment.

So the theorem that is the culmination that we're aiming for is that the GCD of two numbers is an integer linear combination of the two numbers. That is, the GCD of a and b is simply sa plus tb, where s and t are integers. And what the pulverizer enables us to do is given a and b we can find s and t. In fact, we can find s and t virtually as efficiently as the Euclidean algorithm. It's just by performing the Euclidean algorithm and keeping a track of some additional side information as it progresses.

Now a corollary of this fact is that we now know that the-- if we want to characterize the linear combinations of a and b, they're precisely the multiples of the GCD of a and b. That's because we know that every factor of both a and b divides any linear combination of a and b. And therefore, the GCD, which is a factor of a and b, divides any linear combination.

So that means that any linear combination is a multiple of the GCD of a and b. Of course, once we know the GCD of a and b is itself a linear combination, it means that you've got all of the linear combinations by taking multiples of the GCD.

How do we get s and t? Well, the basic idea is that we're going to find these coefficients by applying the Euclidean algorithm. And as we go, we're going to be calculating coefficients. And eventually, when we're all finished, we'll wind up with the s and t.

Specifically, let's remember the Euclidean algorithm starts off with a and b. And then it has two registers, or numbers, x and y, that it keeps updating. And the invariant is that the GCD of the x and y, that are being continually updated by the Euclidean algorithm, stays the same. It's always the GCD of a and b.

So what we're going to do is just keep track of coefficients, call them c, d, e, and f, such that the x, whatever we're up to, we know how to express as a linear combination of a and b. And the y, whatever y we're up to, we can also express as a linear combination of a and b.

So we're going to be keeping track of these four coefficients, c, d, e, and f that have this property. This property is going to be another invariant of our extended Euclidean algorithm, or pulverizer.

Well, how do we get initial values for a c, d, e and f? Well, that's easy. At the start, x is a. And so, c is 1. And d is 0. Because a is 1a plus 0b. Similarly, y is 0a plus 1b. So we know what these values of c, d, e, and f are at the start of the algorithm.

The question is how do we update them? Well, how does a Euclid work? Well, remember, at the next step, the value of x is the old value of y. So if I had the old value of y as ea plus fb, then I clearly have the next value of x as the same linear combination that y had previously.

What about y next? Well, at the next step, the value of y is simply the remainder of x and y. Well, the remainder of x and y, remember, is just x minus the quotient times y where the quotient is the quotient of x divided by y.

So this is equal to the remainder of x and y. And that means that since I also have x expressed as a linear combination, this x minus qy is simply this linear combination for x minus the quotient number times the linear combination for y. Well, the difference of two linear combinations is a linear combination. So just combining coefficients what I discovered is that the way to express y next as a linear combination of a and b is just to combine the previous coefficients, c, d, e, and f with the quotient in this way. And that's all there is to it.

Well, let's work out an example to see how it goes. Suppose that a is 899 and b is 493. These were numbers that we had previously applied the Euclidean algorithm to. So now, what we're doing is observing-- I'm going to begin by calculating the remainder. But this time, when calculating the remainder, let's keep track of the quotient.

So I'm going to find the remainder of 899 divided by 493. It's 406. And the quotient is 1. That is 899 is 1 times 493 plus 406.

What does that tell me? Well, 406 then is-- remember 899 is a and 493 is b. I'm discovering that the first remainder, 406, is 1 times a plus minus 1 times b. So now, I have that first remainder expressed as the desired linear combination of a and b.

Well, what's next? Well, now that I've got 406 and 493, I'm supposed to take the remainder of 493 divided by 406. Well, that's 87. In fact, 493 has a quotient 1 times 406 plus 87. So that tells me that 87 is this number minus that number. 87 is 493 minus 406.

Well, remember, 493 is b. So 87 is 1 times b minus 1 times 406. But wait, look up here. 406, I know how to express it as a linear combination of a and b. So let's replace the 406 by 1a plus minus 1b. And what I'm going to wind up with-- remember, it's a minus minus, so I wind up contributing an a and an extra b. And I wind up with a minus a plus 2b. Said that wrong. The a is getting negated. But you can check my algebra.

So there we are with the linear combination that expresses the next remainder, 87. All right, let's continue.

After this, what we're supposed to do is find the quotient of 406 by 87 and the remainder. So when you divide 406 by 87 you get a quotient of 4 and a remainder of 58, which means the remainder 58 is 406 minus 4 times 887.

But now, looking above, I have the coefficients of 406 for a and b. And I have the coefficients for 87 for a and b here. And so, I have to multiply those by 4 and add them. I wind up that the way to express 58 in terms of a and b is 5a plus minus 9b.

And next, I'm supposed to find the remainder of 87 divided by 58. The quotient's 1. The remainder is 29. And that means that 29 is 1 times 87 minus 1 times 58. Looking back, I see how to express 87 in terms of a and b and 58 in terms of a and b. I can just combine those expressions to wind up with 29 is minus 6 times a plus 11 times b.

Next, I have to take the quotient of 58 divided by 29. Well, the quotient is 2, but the cool thing now is the remainder is 0. That's the stopping condition for the Euclidean algorithm. It means that the answer is 29. There's no remainder anymore. So the GCD of 29 and 0 is 29. The final GCD, then, we finished is 29.

But look what we got. In the last step I had expressed that GCD as a linear combination of a and b. And that's the pulverizer. I've just figured out that possible values for s and t are minus 6 and 11. And this is a perfectly general procedure that will always give you coefficients s and t that express the GCD of a and b in terms of a and b.

Now, sometimes it's technically convenient be able to control which of the coefficients are positive and which negative. Clearly, if you're going to combine a and b that are both positive numbers and wind up with a smaller number by adding multiples of them, one of those coefficients has to be negative.

So in this case, we had the coefficient of 89 was minus 6. And the coefficient of b was 11. And suppose that I wanted, though, the first coefficient of a to be the positive number and the other one to be negative. How can I do that?

Well, there's a pretty trivial little trick for doing that. It's ingenious, but it's immediately verifiable. How do I get a positive coefficient for 899? Well, there's a general way to get new coefficients.

If you look at minus 6 899 plus 11 493, if I add any multiple of 493 to the first coordinate, and I subtract the same multiple of 899 from the second coordinate, all I'm doing is adding 493 times k times 899 to the first term. And I'm subtracting 493 times 899 times k for the second term. They cancel out.

So this linear combination is going to be the same as that one. It's going to be the same GCD. But now, by adding in any multiple-- by the way, k could be positive or negative-- of 493, I can make the first coefficient as big or as small as I like. In particular, if I want it to be positive, might as well take the smallest value of k that works, which is 1.

So if I let k be 1, I discover that I add 493 to minus 6. I get 487. And I subtract 899 from 111 and I get minus 888. And there we are with another expression for-- this time s is 487 and t is minus 888. And the second one is negative and the first one is positive.

It's going to turn out that this little trick will enable us, in the next video, to come up with a general solution to the *Die Hard* bucket problem, which is fun. But let's finish up the current story.

And the remark is that the pulverizer is really another very efficient algorithm, exactly the way the Euclidean algorithm is efficient. It's basically got the same number of transitions when you update the pair xy to get a new pair, y remainder of x divided by y. So it's taking twice log to the base 2 b transitions. So it's exponentially efficient. It's working in the length and binary of the number b.

Of course, there's a few more additions and multiplications per transition for the extended GCD, or the pulverizer, than the ordinary Euclidean algorithm. So big deal. It means that the number of total arithmetic operations of adds and multiplies is proportional to the log to the base 2 of b.

I said here 6. I think it's actually like 10. But the main thing is it's a small constant times the log to the base 2 of b. The pulverizer is a very efficient algorithm as well as the Euclidean algorithm. And those are going to be crucial facts that we'll build on.