DENNIS FREEMAN: Hello and welcome. So as I said last time, the important part of doing this class is doing the homework. We have a relatively complicated new system for how to do the homework, so I apologize if there's a couple of kinks in it. I appreciate the feedback pointing out the kinks. We will try to fix them. And we won't hold anybody responsible for our kinks, just so you know.

So the kinks include there's two parts. There's the tutor problems that are automatically graded. And if you think the tutor didn't give you a just answer, please just let us know and we will take care of it, provided of course that your answer is just. And for the problems that will be graded by a human, we would like you to read the solution so that you understand what went wrong, if anything, in your solution and perhaps an alternative approach. Most of the engineering design problems will be solvable in multiple ways. So, it's good for you to see other approaches.

To encourage you to do that, we're asking you to identify any errors that were in your original submission. So the importance of that is that Homework One was due yesterday at 5:00 PM. And if you want to recoup points that you lost, you must submit marked up version of the same thing that you submitted yesterday by Friday 5 PM, by tomorrow 5 PM.

You are welcome to mark up a photocopy of the paper and give us a paper mark-up. If you do that, it's due in at the beginning of recitation. If you want to scan it, you can send us a scan anytime up to 5:00 PM. Questions? Yes.

AUDIENCE: Can I just email you the scan?

DENNIS FREEMAN: No. Enter it into the web site. It's much easier for me. The graders will all be able to see it that way. If you send the email it's much harder. It's easier if we have everything in one place. If there's a problem with the tutor, then send it to me by email and we'll fix it. Other comments or questions? Yeah?

AUDIENCE: [INAUDIBLE] Do you ever get them [INAUDIBLE]?

DENNIS FREEMAN: Yes, eventually. Now, what hasn't been completely worked out yet is whether what you get back will be paper or [? bits. ?] Because we'll probably end up doing all the grading via [? bits, ?] because it's just easier to communicate with all the graders via [? bits. ?] So that's not been resolved yet. We're still going to work out with the graders what's the easiest way to synchronize grading the electronic and dealing with it.

So you will get feedback but you won't necessarily get paper feedback. Other comments?

OK. So last time, we started to think about how to think about discrete time systems. The thing you were supposed to get out of last time is that there's different representations for discrete time systems. We're interested in all of them. We're interested in all of them because of the strengths and weaknesses of each. We looked particularly at difference equations, which are mathematically precise and concise. That's very good. We looked at block diagrams, that was good because we could trace the flow of information through the network.

And we looked at operator representations and they were nice. And we will see further today how that's nice. They are nice because they allow us to think about a system in terms of polynomial mathematics. So what I want to do today primarily is develop that latter theme.

Keep in mind that one of the important features of the block diagram was the arrow notation. It told you what causes what. It let you make a hardware realization of a system.

An important point from last time, which we just ended on at the end of the hour, an important point was that when you're thinking about the signals, there's a higher level abstraction that is also important. In addition to knowing which is the input to the R operator and which is the output of the R operator, there are some kinds of systems that just behave categorically different from other kinds of systems. And they are indicated here.

This system has the property that is easy to write a recipe for how to compute y given x. How do you compute y given x? Well, you take x, copy it, invert it, shift it to the right. Add the copy to the inverted shifted. Easy to make a recipe to do that.

It's not so easy to make a recipe here. Right? So there, so if you think about how you would try to write that in an operator, you could say, y. What is y? y is going to be x plus a right-shift of y. So you could say, well that's 1 minus a right-shift of y equals x. That's not a recipe for computing y from x. That's a recipe for computing x from y. But we don't know y. Everybody see? That's completely different.

And so it's that difference that makes this kind of a network, this kind of a system interesting by comparison to that one. And the distinguishing feature that makes that behavior occur is feedback. When you have feedback, feedback means, for some signal in the system, we would say the system has feedback if there is some signal somewhere whose output at time n depends on it's same, the output of that same signal at a previous time. If that's true for some signal in the system, then we say the system has feedback.

The implications of feedback are profound. They mean, for example, that a transient input can give rise to a persisting output. And we took a look at how to think about that in terms of the block diagrams. You can see that if you had even a signal that only lasts for one sample, a signal-- a system that has this property of feedback could generate a response that lasts for a very long time.

In fact, the response at time 0, the only way that could happen is through the straight through path. The only way you could get an output at time 0, given that the input is non-0 only at 0 is through the straight through path. But then you could get an output at time 1 if you cycled around once. This is the idea that a signal somewhere depends on a prior value of itself.

And you could get a response at time 2 by going around the loop twice, three, 3 times, 4, four times, et cetera. And in general, the response could persist forever.

That's such an interesting behavior. It's such a fundamental property of these systems that we want to characterize systems by thinking about them that way. So in this particular case, if we think about sort of the most simple form that feedback could take, this is sort of the simplest way you could imagine. This is kind of the simplest system that you could imagine that has feedback.

And you can see that the kinds of responses that you can get depend on this number p-naught. And that p-naught characterizes the family of responses that you could get. So for example, if p-naught were a number less than 1, you could get a decaying geometric sequence. If p-naught were 1 exactly, you would get a response that never decayed or grew. And if p-naught were a number bigger than 1, then the response could grow infinitely. So that's a very fundamental characteristic. And we call that number p-naught the pole. OK.

Everybody's happy with that? That's very simple-minded.

So we're going to characterize-- so the point is, there's a higher level of structure in systems that we would like to capture. We're going to call that higher level feedback. It means there's a cycle in the block diagram. And it means that you can get characteristically different kinds of responses then result when you don't have feedback. In particular, simple systems will have this behavior of a geometric sequence characterized by a pole.

OK, just to make sure that everybody is with me, how many of the following unit-sample responses can be represented by a single pole? OK. Before you answer that question, talk to your neighbor. Don't even bother trying to think about the question. Look at your neighbor. Say hi. And then try to answer this question.

[AUDIENCE CHATTER]

DENNIS FREEMAN: OK. What's the answer? Everybody raise your hand. Raise some number of fingers equal to the answer for how many of those responses could result from-- how many of those systems could be represented by a single pole? OK. A little variety. There's about 80% correct.

Tell me a simple one. So tell me one that can definitely be represented by a pole. OK. Upper right. Because it looks just like one that I showed you before. What's another simple one that can definitely be represented by a pole? Excuse me?

AUDIENCE: Right underneath.

DENNIS FREEMAN: Underneath. OK, so these are things that were on the previous slide. That should be pretty easy. Any others? Yes?

AUDIENCE: Top left.

DENNIS FREEMAN: Top left. Why? There's none like that on the other side?

AUDIENCE: Because it's oscillating. It's changing sides and it's pole is negative but because it's converging to 0, the pole, it's magnitude is less than 1.

DENNIS FREEMAN: So what would be the pole for the upper left?

AUDIENCE: Some p-naught-- so that is like negative 1 p-naught plus p-naught less than 0.

DENNIS FREEMAN: So can you figure out what p-naught is? Back in the back? Any? And so what's p-naught? For the upper left, what is the value of p-naught?

AUDIENCE: 1/2.

DENNIS FREEMAN: Minus 1/2. OK. Maybe. Not really. But yeah. So if it were, if p-naught were minus 1/2, let's think about p-naught is minus 1/2, what would that do? OK. If the first answer were 1, then the second one should be minus 1/2? What should the third one be? Plus 1/4. What should the next one be? Minus 1/8.

So it doesn't quite look like a minus 1/2. It looks more like minus 3/4. Yep. Having made the graph myself, I know that the answer was minus 0.8. OK. You wouldn't necessarily have any way of knowing that.

OK so three of them are, how about this one? Yes? No? Completely uncommitted? My neighbor didn't know? My partner's an idiot?

So this is geometric, isn't it? Right? The only thing you see there is like 1 and 1/2 and 1/4 and stuff like that, right?

Blank stares. Yes?

AUDIENCE: So it's not [INAUDIBLE], so it's not diverting, so it can't be [INAUDIBLE] unit step.

DENNIS FREEMAN: If it were responding to the unit step, that's absolutely true. So if I'm thinking about the unit-sample response, then the unit-sample response always looks like p-naught to the n, n bigger than or equal to 0. And there's no way you can choose a p-naught that fits that lower left hand.

How about this one? Same problem. It's alternating with the same problem. Right? So the only response-- so you can't do those two. The answer was three. The only kinds of responses you can model with a single pole can be shown here. That's not quite true. Let me retract that slightly.

Here are some-- and I will make this more complicated and about two pages of notes. Here's all the real values of p-naught and the associated behaviors. In two pages, we'll get to the complex values for p-naught.

So here's all of that the responses that can be associated with a single real valued pole. So you can get, as somebody has already said, you can get the idea of diverging with alternation, converging with alternation, converging monotonically, diverging monotonically. Those are the behaviors you can get with a single real value pole. OK.

That's what can happen with a single real valued pole but systems, there are more complicated systems. And when you have a more complicated system, you can get a more complicated response. And one example is shown here.

So here a system. Here's the unit-sample response. Is this something that can be represented by a single pole? No, because it doesn't fall into one of those canonical shapes. Right. In fact, this response has the property that it grows and then decays. But it only does the grow and decay once, so it's just not one of the canonical forms.

So now there's something different going on here and we'd like to understand what that is. So we cannot represent that by a single pole.

However, this is where the polynomial stuff becomes useful because I can think about tricks I can do with the polynomials, which you all learned in high school so you all know how to do that. That we can think about the system in terms of, we can think about this complicated system in terms of simpler systems that can be represented by one pole.

So for example, one of the more powerful things you can do with a polynomial is factor. If you factor this polynomial-- OK, so this is a simple representation for that right? I can just read that straight off. So I can make this representation just reading it straight off the block diagram. I can turn that into an operator type representation here. So I operate on y with this mass to get x. It's backwards feedback, that's what happens.

But the second order polynomial in R can be factored into two first order polynomials in R. And not surprisingly, that then leads to a representation of the more complex system in terms of two simpler systems.

So in fact, I can write the original system. If I factor it, there's an equivalent representation. Equivalent means provided that all the systems start initial rest. Initial rest means provided all the delays started out at 0. Putting in the normal caveats, there is an equivalent representation to the complex system. In fact, there's two equivalent systems shown here. Each corresponds to factoring. And it corresponds to doing which factor first.

So in the first realization, it says that generate an intermediate signal by thinking about the factor that has a base of 0.7. Then cascade that with another system that has a base that looks just like a canonical first order system, except that the base now is 0.9. And presumably, if you cascade those two things, you will get an equivalent response. And the order didn't make any difference.

So that's a way we can think about the response of a more complex system in terms of first order responses. So that's nice. And that gives us then-- so we can then use the idea of the Taylor series expansion to think about the response of each of those first order systems, so we can replace the response of the canonical first order system with the infinite feedforward realization-- remember from last time, the first order feedback system can be equivalently represented by an infinite order of feedforward system-- so that the unit-sample response here.

So this system, excuse me, this first order feedback system can be rewritten as that infinite order feedforward system. This one can be written that way. And then again, we can use polynomial math to figure out the response of that more complicated system. All we need to do is multiply and then collect terms. You all know how to multiply polynomials, therefore you all know how to solve second order systems. OK.

That's the advantage of converting one representation, a system representation, into a polynomial representation because you already know all about polynomials. Since you already know how to multiply polynomials, you already know how to cascade systems.

The only difficulty here is bookkeeping, not missing a term. So there are graphic aids to that. Think about the block diagram that would correspond to the infinite feedforward realization of the left part and the infinite feedforward realization of the right part. So I'm thinking about this system and that system being represented by this infinite system and that infinite system. And now just making a block diagram realization of this and a separate block diagram realization of that and cascading them to get this. And now the bookkeeping is trivial.

What's the only way you can get a response if the input is a unit-sample signal? What's the only way you can get a response at time n equals 0? What's the only signal path that generates a response of n equals 0? Say it louder?

AUDIENCE: The top right left.

DENNIS FREEMAN: The only signal path is to go through here, because if I go through here, I introduce a delay. This introduces even more delay. This introduces even more delay. The only way I can get from the input to the output with no delay is to go through the one here and go through the one here. So that that's where that term comes from. The only way I can get from the input to the output with no delay, no R, is through that top path.

If I want to get one level of delay, the only way I can do that is to pick it up here and then not pick it up there or vice versa. There's more ways that I could get two delays. I can pick up two delays all at once on the left with none on the right or vice versa or one in each. That's just bookkeeping, right? It's just showing you how you can think about polynomial multiplication in different ways.

Alternatively, you could represent it in a table. So if you think about multiplying these two polynomials where I've represented the first pole as a in the second pole as b, you can think about polynomial multiplication in a tabular form. So you have to think about the distributive property, the 1 distributes over all of these, then the a distributes over all, the aR distributes over all of these and the a-squared R-squared distributes over all of these. And you can represent each of those distribution operations by a row in the table. And then you can see that the only entries with no R, with R to the 0 is along that diagonal. And then the R terms are here, the R-squared terms, the R-cubed terms, et cetera.

And you can see how that would all sum to give you something that's not a geometric. OK, all those things, just bookkeeping. Right? All I'm trying to say is that there is a way that you can think about the second order system which does not generate a simple geometric sequence by thinking about the cascade of two systems that each do generate a unit-sample response that is geometric.

Nevertheless, all of this is bookkeeping. It's still the case that this is a complicated shape. And you still have to do a fair amount of work to figure out what that complicated shape is. We can use polynomials one more time to make it conceptually simpler.

We can do something that we call partial fractions. You can take the expression that we found for this complicated system, factor it just like we did before. But now, according to the method of partial fractions, you can represent the reciprocal of a quadratic as the sum of two terms, each the reciprocal of a linear function in R. So you can always do that. That's a property of polynomials, right?

So what that says is that there's an equivalent representation for this complicated system that looks like this. So backing up, so we showed that by factoring, we take this and turn it into this. By partial fractions we turn it into a sum difference. What's the difference? So now we think about that difference here.

This term can be generated by multiplying 4.5 times a canonic first order term. This can be generated by multiplying minus 3.5 times a canonic first order term. And the response adds.

OK, well that's completely new insight. We've taken what looks like a very complicated response from a system that is not a canonical form. And we've turned it into something that does look like several canonical forms, just like we did with the cascade, but now we can see that the answer is the sum of two geometrics. Everybody's with it?

So it's even better than being able to cascade, wherein the cascade, we have to do some complicated bookkeeping in order to figure out the responses. By thinking about it with partial fractions, it's the simple sum of two simple things. So then if you know properties of each individual one, life it rattles for a long time or it decays quickly, you can instantly say something about the combined response. Does that follow?

So then this response, which I introduced by saying it's complicated, it's not really complicated. It's the sum of two geometrics. It goes up and then down simply because the decay constants for the two different geometrics are different, that's all.

The point is that this is a very different kind of representation. Many things-- so you could do a lot by just taking a difference equation and just factoring it and moving terms around and figuring out different ways to do the sums and so forth to try to simplify your life. You can do the same with block diagrams. You could shuffle around the blocks and try to make things simpler.

This idea of turning the top system into this is trivial to see with polynomial manipulations and relatively and not at all straightforward to do by simple block diagram manipulations. It follows directly from the polynomial realization. That's what we mean by a more powerful representation. It's very easy to see how we got there if you represent the system by a polynomial. It's much less easy to see how we got there by thinking about it as a block diagram or as a difference equation. OK? Makes sense?

Because they're so important, we would like shorthand ways of finding poles. As soon as you know the poles of a system, then you know the geometric sequences that characterize unit-sample response. And for at least a broad class of systems, we're going to be able to think about the combined unit-sample responses of the system as some weighted sum of individual contributions.

So that makes the idea of finding the pose very important. There's a very straightforward way that we've sort of already alluded to. If you had this complicated thing and you wanted to find the poles, what you should do is try to coerce it into the canonical form.

You remember the canonic form for a single pole system was this sort of representation. So you have x coming in. Coming out as y and feeding around that way. That's the canonical form for one pole system. And that had a system function of the form y over x is 1 over 1 minus p-naught R.

So if you can coerce all the parts to look like 1 over 1 minus p-naught R then according to that, here's a pole and there's a pole just by canonic forms and matching up the terms in a canonic form, right?

That requires some sort of graphic memory or something graphic ability to take one form and turn it into another form. A simpler rule is to say, look, just replace all the Rs by 1 over z, rewrite things, and the z's turn out to be-- the values of z that are the roots of the denominator turn out to be the values of the poles. That's very straightforward to see just by looking at the canonical form.

If I have the canonical form where I know that the pole is p-naught and if I replaced the R by 1 over z and cleared the fraction, I would end up with z over z minus p-naught. And now if I ask what's the poles at the bottom, what's the roots of the bottom? For what values of z is the denominator equal to 0? The answer is clearly z equals p-naught. It's just a trick to turn a system function in terms of R into a root-finding problem. So rather than having some program that knows how to write canonical forms, you can use any root finder to find the values of the poles.

OK so make sure everybody is with me. Consider the following system, blah, blah, blah. How many of the following are true? Answer looks like it ought to be a number between 0 and 5 probably. So talk to your neighbor and figure out a number between 0 and 5.

So how many of the statements are true? Hands?

OK, participation is a little small, but it's like 95, 93, something like that percent, except people keep changing their number of fingers. That makes it confusing. OK.

How do you think about the first one? The unit-sample response converges to 0, yes or no? How do you think about it?

Yes because they all do. True or false. Yes. Because there's something special about this one. What is special about this one that convinces you that that's true or false? Yes.

AUDIENCE: It has two poles both of which are magnitude less than 1.

DENNIS FREEMAN: It has two poles, both of which are magnitude less than 1. That's precisely right. So if we simply-- if we just do one blackboard full of math. So we take this. Turn it into a polynomial form. Turn it into a functional form. Turn it into z's. Everyone's happy with that?

Now you may have made slight algebra errors in what you were doing, but the procedure is very easy, right? We see that we get two poles. Where are the poles? Negative 1/2 and plus 1/4. They're at the roots of the denominator. So is the unit-sample response converge to 0? It's the unit-sample response. Unit-sample responses are going to be the sum of two pieces from the partial fraction argument. And so it will depend on the positions of the poles. Both poles are inside the unit circle. Both poles have magnitude less than 1. Therefore, they converge, therefore the first one is right.

Poles at a 1/2 and 1/4, obviously wrong, right? Obviously wrong, right? There are two poles. None of the above. Obviously wrong. OK. So the answer is 2. OK? OK.

One more example problem. Favorite computer science problems is the Fibonacci problem. Fibonacci was thinking about population growth when he formulated the problem. The idea was what if you have a pair of rabbits that takes one cycle to grow up. And every cycle thereafter, has a pair of baby rabbits. Then the next cycle, there is a pair of baby rabbits.

Then they grow up. But meanwhile mom and dad had another pair of rabbits. But then those grow up. And now we have more. And then they grow up. And then they grow up. And then they grow up. And you get lots of rabbits. OK. So that's the idea, right? Question? What are the poles of the Fibonacci system?

AUDIENCE: The poles, why isn't that-- [INAUDIBLE] to find the poles [INAUDIBLE] to solve for a denominator [INAUDIBLE]

DENNIS FREEMAN: Why is a hard philosophy question. I'm not sure I can answer philosophy questions. I guess I have a PhD, maybe I can. So it has to do with partial fractions, right? It's the bottom. The bottoms are special. And in fact, in about a week, we'll talk about what's the tops. Not surprisingly, the tops are like an inverse bottom. What a deep insight.

So if you were to run signals through the system backwards, the tops look like bottoms and the bottoms look like tops. And there's some really useful things that you can derive from that.

OK what's the answer to the question? What are the poles in the Fibonacci system? Give me a number between 0 and-- 1 and 5. Raise your hand.

OK, how many are you guessed? OK. That's what I thought.

OK so, are there poles in a Fibonacci system? How would you convince yourself there are or are not poles in a Fibonacci system? Yeah?

AUDIENCE: You can [INAUDIBLE] as a difference equation, so [INAUDIBLE] some visual representation.

DENNIS FREEMAN: So you have to start somehow. Somehow you want to translate that graphical problem into a difference equation or an R expression or something. So one way to think about it is a difference equation of this form. At time n, add some number of rabbits that somebody just plunked in. That's my input. In the example I showed by magic, by spontaneous creation, there was a pair of rabbits at the beginning that didn't obey the rules for that were true for the rest of time. That was x.

And y of n, also you have to add in the two previous values. That's kind of a rule for the Fibonacci system. And that has the form that we've been talking about. It's something that has just adder's and delays and stuff like that in it. So we can write that as a functional. And we get a second order denominator so we see that there are two poles.

And if you find the roots by the z method, you get these funny numbers. And they're indeed funny. You get the golden ratio and you get the reciprocal of the golden ratio, the negative reciprocal of the golden ratio.

So that's really bizarre because we end up with this whole pole, 1.618. Tell me something about the response of a system that has a pole at 1.618.

AUDIENCE: It [INAUDIBLE].

DENNIS FREEMAN: Diverges. What about this one at minus 0.618?

AUDIENCE: Oscillate.

DENNIS FREEMAN: It oscillates. So you take diverging plus oscillating and what do you get? You get a sequence of integers. OK. That's pretty bizarre, right? You take diverging and oscillating and you get 1, 1, 2, 3, 5, 8, 13. That's pretty weird.

Also notice that it doesn't behave the way we think about it when we think about computer science. There's no r-squared growth of anything. How many operations is required to compute the output at time 119?

AUDIENCE: [INAUDIBLE] operations.

DENNIS FREEMAN: Depends on what you mean by operations, that's a good point. So if you used a modern laptop, how many steps would the laptop have to take? You have to know something about what's inside the laptop. These days, exponentiation is pretty simple. So you can think about-- so all you need to do is raise phi to the n an add minus phi the minus n and scale them and add them.

So it's a different way of thinking about things. That's the point. So you can think about the Fibonacci system as a recursion problem like we did in 601 or you can think about it as the response to an LTI system, a linear time invariant system and you get this kind of a way of thinking about things.

OK one more thing to talk about today. And that is, what happens when things are complex? You can make a terribly small change to the system functional. Here all I've done is change one sign from the Fibonacci system and the result is complex roots. So what's it mean to have a system that has complex poles?

So complex poles, one of the interesting things about polynomial math, complex numbers work like any other number. Not surprisingly then, complex poles in a linear system work sort of like any other pole. The fundamental mode associated with any pole is p-naught to the n. The fundamental modes corresponding to a complex pole have complex values. And that's shown here. We can write-- so we can think about that system polynomial that I showed in the previous slide. This one. We can write that in a factored form.

But now, the roots are complex numbers. This is a complex number. There's complex numbers all over the place. Who cares? Complex numbers, they're just numbers. That's the nice part about algebra. So we don't need to worry about that.

Other than the modes are complex. So if you try to plot the mode associated with one of the roots and the other root, they have real and imaginary parts. So here I'm showing the real part in blue and the imaginary part in red. So each of the fundamental modes has a real and imaginary part. That's the only complicated thing.

It's sometimes easier to visualize the modes on a polar plot. And that's illustrated here. These plots look complicated. If you think about what they look like in polar coordinates. If you think about this p-naught to the n, but now p-naught is complex, just plot the number p-naught to the n on a complex plane, it's pretty simple.

So in this first case, we have this pole at e to the j and pi over 3. Gets raised to the n-th power. And we use trig and we can come up with some expression for that. But forget trig, just remember the Euler relationship. And that's equivalent. We can just use this form to say well, if you had 0, e to the 0 is 1. So I'm there.

If n were 1, then it would be e to the j pi over 3. Well, that's there. You go up by 60 degrees. If n is 2, you're over here. 3, 4, 5, 6, 7, 8, 9, 10. So you can visualize how the root is behaving, the complex root, by looking at a complex plane. That other root works just the same way, except it's negative so it spins around the other way.

So it's convenient to watch the behavior of complex roots on a complex plane. That's not too surprising. What's slightly more surprising is the idea of a real system that's made out of adders and delays and that kind of stuff generating a complex response. That's a little weird. How would that happen? Does that mean every time I build a delay, I need to have an output that is imaginary? Or does it sort of just squirt it out into some orthogonal space? Or how do you think about that?

So it's a little weird to think about the response of a real system coming out complex. And they don't. Systems with real value coefficients of the type that could be represented by a difference equation, think about a difference equation. If you require that the coefficients are real valued, that puts a constraint on the kinds of difference equations that you can have. Even though the roots may come out complex, if you started with a system that could be expressed with a difference equation whose coefficients were all real, then algebra will conspire to make the sum of the modes be real. It sort of all that could happen. Right?

So we're expecting that if it were a physical system, if it could be represented by a difference equation with real value coefficients, how on earth could a real valued input generate a complex valued output? That doesn't make any sense. And it doesn't happen. And that's because things conspire.

So this is just showing long-winded math for those of you who like long-winded math. The idea is that these coefficients conspire as complex numbers so that for every integer value of n, the answer will be real valued. And what's showed in the bottom is the unit-sample response for that system. And you can see everything's real valued now.

So last question. Here is the response of a system with poles, two poles at those complex numbers. How do you think about the values R and omega? So assume that the poles are at the polar form of the pole location is magnitude R, angle plus or minus j omega. How do you think about R and omega?

So let's say that I plotted the response here, can you figure out by looking at that, what is R? Can you estimate R? R is approximately 1/9 0.9, approximately 0.9. So if it were 0.9, how's it go up? Hmm.

So it's going around in that circle right? So as it's spinning around in a circle, it's going up and down. As it's spinning around the circle, the R is also spiraling in. So the R, R to the n. Right? If we think about pole to the n. The key is always pole to the n. If we think about pole to the n, then it's R to the n, e to the j omega n.

The R part, R the n is making it spiral in or spiral out, one of those. And the omega part is making it go up and down and up and down. So the R here is actually 0.97, because I made the plot so I know what it is. So R is 0.97. That has to do with sort of a gentle decay across cycles.

So you could figure that out for example by finding the ratio of that height to that height. Sort of, how much did the decay when it came around once? So R is 0.97, what do you think omega is? Yeah?

AUDIENCE: [INAUDIBLE]

DENNIS FREEMAN: Around 0.5. So what you need to think about in omega is if you go n steps and each advances the angle by omega, how long does it take to get to 2pi? Right? It will go around the circle once when the radiant angle, when the angle goes increments by 2 pi. Every time, every step, the angle is incremented by omega. How many steps does it take to get to 2 pi?

That takes-- if I started here and went to there, I get 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12. Takes about 12. So 12 omega is 2 pi. And so omega is approximately, well it's exactly, well, no it's approximately-- this should be approximately 2 pi over 12. 6 over 12 by blackboard math is 1/2. OK.

OK there's one more slide. But that will turn out to be a good question for you to review as preparation for the exam.