MIT OpenCourseWare
http://ocw.mit.edu

18.085 Computational Science and Engineering I, Fall 2008

Please use the following citation format:

Gilbert Strang, *18.085 Computational Science and Engineering I, Fall 2008*. (Massachusetts Institute of Technology: MIT OpenCourseWare). http://ocw.mit.edu (accessed MM DD, YYYY). License: Creative Commons Attribution-Noncommercial-Share Alike.

Note: Please use the actual date you accessed this material in your citation.

For more information about citing these materials or our Terms of Use, visit: http://ocw.mit.edu/terms

PROFESSOR STRANG: OK, let's start with a review and preview. I put a P up there because we're really looking into the Fourier part that just started this morning. And there'll be some homework from these early sections about the Fourier stuff, so we maybe we should just do a few of those problems or discuss here today. Just in advance. Can I say one thing about MATLAB and the MATLAB homework first? And maybe open a conversation about it? So there's really two different problems that I'm personally quite interested in. Two model, I'll say model problems because they're there for regular polyons in a circle. And I'll draw an octagon again, so M sides. And I'm interested in if M goes to infinity. And I'm interested in two different problems. So one of them is our MATLAB problem, minus is Laplace's equation. What was it, four? With u=0 on the circle. OK, so that's our problem, totally open for discussion. How many have started on that? Oh, good. OK. Well, then you all know more about it than I. And that's great. I'd be happy to learn. So have I said everything there? Yeah, we've got Poisson's equation inside. We've got u=0 on the circle, so the problem's well defined and the solution should be one minus x squared minus y squared. So that's the correct solution.

Maybe I can also tell you about the second problem that I'm interested in. Because it hasn't come up in class but it's very important, too. It would be the eigenvalue problem. So this is problem number one, the steady state problem when you've got a source and you want to find out the temperature distribution. The problem number two would be the eigenvalue problem, $-u_{xx}-u_{yy}$. I take those minuses so that the eigenvalue will be positive. So that's what the eigenvalue problem might look like. And again let me say, with u=0 on the boundary. On the third one. OK, so a person would say this is Laplace's eigenvalue problem because we have Laplace's equation. We've got eigenvalue. As always, it's not linear because we have two unknowns, lambda's multiplying u. And we have boundary conditions, and this would describe the normal modes, for example, of a circular drum. If I had a drum, or a polygon drum. So maybe I connect to actually build the drum, I might fold in the sides there and have a polygon. And again, I hope that the eigenvalues of the polygon, this equation in the polygon, which are not known, by the way. To the best to my knowledge, we know it only for M=3, which would be an equilateral triangle, and M=4, which would be a square. And those eigenvalues, because of Fourier or something are humanly doable. But I think five on up is, I may be wrong about six I'm not sure about M=6, a hexagon sometimes gives you enough help. But beyond that you're on your own. With finite elements to help you.

So there's a whole sequence of eigenfunctions, u, eigenvalues, lambda, just the way there were in one dimension. And on the circle they involve Bessel. That's where Bessel showed up. He figured out the functions and they're not especially nice

functions. But they're studied for centuries. Bessel functions come into that. But, here I have the same question. I mean, let me just say, for me this could be a UROP project if anybody was an undergraduate, or it could be a project over January or something. I'd like to know something about what happens as M goes to infinity as the polygon approaches the circle. So I'm hoping maybe on the homework that comes, if it's not too difficult, and maybe it's not, to let m go up a bit. There is one thing. That the code we're working with is linear elements, right? We're using linear finite elements. So we're not getting high accuracy. So I would really like to move up to quadratic elements, at least, you remember quadratic elements would be ones where - well, let me draw the one that we've drawn in class before. We only have to look at one triangle and then we cut it up into triangular elements by taking some pieces here, taking the points above, which I hope are now correct on the website. Connecting those edges, and then connecting these. Is that right? Is that our mesh? So that mesh is controlled by N. One, two, N points. Also, N is going to have to get large too, to give me accuracy. And another way toward more accuracy is, instead of linear elements, second degree. So do you remember I wrote those down? Let me take that little triangle out here as a bigger triangle. It would look something like that, I guess. The second degree elements have those six mesh points. You remember I drew those but we didn't really have time to get further with them.

The trial functions phi, which are one at a typical mesh point and zero at all the others, they are computable. We're up to second degree, so it's a little, second degree things, then the first derivatives, which come into the integrations, are linear. And not constant. So a little bit harder. But finite elements, linear or quadratic. Or higher. Could be used for this problem. As we know, and for this problem. What I wanted to add, that I've not mentioned in class, and I think we may just not get a chance to do it, is what does the finite element method look like for an eigenvalue problem? Because eigenvalues are highly important. That's the different way to understand. There's the matrix K and its entries. But then there eigenvalues. And you might think that, what do you think is the discrete eigenvalue problem copying this one? Here's my point. Your first guess would be, well this is like K, right? This is like KU, right? (K2D)U, I should call it, maybe. Well, I'll call it K, because K2D I have specifically reserved for the Laplace stiffness matrix on a square mesh, square mesh with triangles, the K2D. That was one specific matrix for one specific mesh, and here we have a different mesh. So I should just call it K. Ok, I think if anybody was going to make a guess, they would say OK, KU=LAMBDA*U. Maybe I'll use capital LAMBDA, because I'm using capital U. Is this the finite element method eigenvalue problem. And if you answered yes, I would have to say, well that's a reasonable answer. But it's wrong. The eigenvalue problem, when I take the differential equation for the Laplace, Laplace's equation, lambda u on the right side, and I go to do finite elements, it produces K. Out of this stuff, out of the weak form, all that stuff. But it produces another matrix on the right-hand side from the constant term, and we have not really mentioned it, it's the mass matrix.

So this, instead of just the identity here, there's a mass matrix. So that is the problem that you could do. I could've made a MATLAB project. I bet I'd do it next fall. right? You guys did the first one, this one. Or you are doing it now. And I'm going to pause in a minute for questions about it, or discussion of it. But this one brings in something called the mass matrix. So let me just say what those are. If I write down the entries in the mass matrix, you'll sort of get an idea of why they are. So what are the entries in the stiffness matrix? K_{ij}, you remember, is the integral of the d phi_i/dx, d phi_j/dx. Plus d phi_i/dy, d phi_j/dy, dxdy, and that's what's you're computing. And that's what that code is computing. And when phi is linear, phi

linear, then slopes are constant. So all you have to do, and what that code in the book is doing, is figuring out what are the slopes. These things are constant, so we just need to know the area of the integration where we're integrating. The area, triangle by triangle. Fine. That's what we're doing. That's what that code is just set up to do. Now, I have to tell you what is M_{ij}, the mass matrix. I just think you don't want to have - we haven't done too badly with finite that elements in here. We did it in 1-D, where we got it kind of straight. And now we're seeing what it looks like 2-D. But I had not really mentioned a mass matrix. So here it comes. The mass matrix will be the integral of phi_i i times phi_j. dxdy. It's the zero order, no derivatives, just plain zero order, as you'd expect from the fact that the term in the continuous part is zero order. So it's this mass matrix that comes in. And maybe we could just look to see which entries will be zero and which will not.

How sparse is it? What does the mass matrix look like? And can we just, let me do 1-D first. So there's a phi, right? There's another one. There's another one. So, what do you think about the mass matrix, one phi multiplied by another phi and integrated? Is it diagonal? No, because each phi overlaps its two neighbors. So tell me what kind of a matrix m is going to be? In 1-D. Tri-diagonal. It'll be tri-diagonal. Now, so was K. So K and M actually have non-zeroes in the same places. the same sparsity pattern. But, of course, not the same numbers in there. K had minus ones and twos and fours and minus ones. What can you tell me about this tri-diagonal matrix? When I integrate that against this, well, again I would do it element by element because this against this, they only overlap here. Right? I'll just draw the one place that they overlap. And what's the point? They're both positive. So the mass matrix is, its rows don't add to zero. Its rows tend to add to one. But it's not diagonal, that's the difference. OK, so I just felt I couldn't feel as though I'd done a decent job in describing finite elements if I didn't describe this. Didn't mention this mass matrix. And maybe I'd better say where it comes from. Because eigenvalue problems, it may come number two, but that's pretty high up the list. So let me tell you where does this mass matrix come from. First, let me tell you about eigenvalues of a, matrix eigenvalues. So the answer was, is this the finite element eigenvalue problem? Only if there's an M there. And now I want to, OK, first of all, what MATLAB command solves that problem? Let's just be a little practical for a moment. What MATLAB command gives me the matrix of eigenvectors, the matrix of eigenvalues would come from eig of what? I'd call this the generalized eigenvalue problem. Generalized because it's got somebody over here.

And it's just K,M. Or of course you get the same answer, well you get the same eigenvalues, I guess the same eigenvectors, yeah, if you or, eig(M^-1,K), of course. If you want to do it with just one matrix then bring M inverse over here. But and M inverse, the inverse of this tridiagonal matrix, is full. No zeroes in the inverse. So everybody would much prefer this tridiagonal tridiagonal one. So that's how MATLAB would do it. And what I want to know is, back in this problem, how close do the finite element guys come, on polygons, come to the correct solution on circles. I'm hoping that for problem one you can maybe keep M and N equal, or maybe four times M or something. And let them grow and see. Well, for example, at the center of the circle, or how quickly do you approach the correct answer one at the center of the circle? I think it's going to be a good problem. Let me open to, so I started out just talking there. What about the MATLAB problem. You made a start on it, is it going? Have you got a graph, maybe, or what's reasonable to graph, to give Peter to look at? Who's done something on that MATLAB problem? Yeah, go ahead tell us all what to do.

AUDIENCE: I made the triangle pi section

PROFESSOR STRANG: OK, right

AUDIENCE: [INAUDIBLE] and I found that the [INAUDIBLE] changes to M.

PROFESSOR STRANG: With M more, I see. So if you just fixed M like eight, and let n get, it didn't change significantly. It wouldn't, of course, converge to the right answer. It'll converge, if it does, to some kind of an answer, for the polygon. Right. That's right. So you know, as I wrote the problem I didn't know whether I dared say let M get increased too, but of course that's the real question. And what happened then? Did error shrink? OK, and now maybe it's possible to see how fast or something that's always--

AUDIENCE: [INAUDIBLE]

PROFESSOR STRANG: Ah. OK, at the center. OK, then I hope for more comment. Let me say one more thing. My theory is that the error at the center is quite a bit smaller than the error closer to the boundary. I would be interested in an error, is it fairly even? Oh, my theory's wrong. It wouldn't be the first time. And maybe because it's linear. Yeah, my theory is more for better elements, like these. I'd be interested to know. Why do I think, why do I have this theory, which you guys are going to prove wrong anyway, but still. After you've proved it wrong, you won't listen to me if I tell it to you. So now I'll tell it. My theory is that the error around the boundary is, there's no error at these vertices, and then there's sort of a going to be an error because the real answer is not zero along here. It's sort of near zero, but not quite. You know, there's an error. So there's errors around here, from getting the boundary wrong. Squaring it off. But my theory is that errors, the boundary stuff, drops off quickly as you go inside. That's why I think, from those, you remember those - well, we'll see them again either today or Friday, those r^n*cos(nx) type things? That cos(n*theta)? Yeah, you remember those are the typical solutions to Laplace's equation. And then so that if, and it has some coefficient, of course, a n. And I look at that, that might be a piece of error. And it's way bigger when r is one and way smaller when r is zero. So anyway, that's sort of my theory. That if you have, like physically. You have a circular plate and you're maintaining the boundary temperature at some sort of oscillation. Like, near one but up and down from one. Then I think further inside, it doesn't know. It hardly knows about that oscillation. This is my theory. That toward the center of the circle it only sees kind of an average boundary temperature and not your little ups and downs. So when M is big, I expect that part of the up and down part to be not so significant in the center. Anyway, now that's my theory.

AUDIENCE: [INAUDIBLE]

PROFESSOR STRANG: Ah, good question. So if we only looked at the center, would it all be the same? I mean, if we're only looking at that one point where it should be 1 at the center, but along the thing, I don't know. If you look at both, and see a significant difference in the behavior I'd be interested. Yeah, yeah. You know, all these problems are things that there's no single solution to.

AUDIENCE: [INAUDIBLE]

PROFESSOR STRANG: The error between one minus r squared

AUDIENCE: [INAUDIBLE]

PROFESSOR STRANG: Oh, right, we've got slope error, too. That's a very significant point. I see, right. So the slope error's in there. Everybody knows, then, everybody in working the problem, I mentioned that the boundary conditions in this piece of pie were zero along here and normal derivative, somehow it got printed du/dh, but that was an accident. It should've been du/dn, dn is zero. So Neumann conditions on this thing and then I was a little scared about that point, but I think phooey on it. It's just, don't worry about it. But what I was going to say. How do you, what do you do to take into account this du/dn=0? This slope condition on these long boundaries? What should you do in finite elements to take account for that? And the answer is, in one nice word? Nothing. Right, nothing. Your finite element method should not, you don't impose any condition along these boundaries. Just use the code as it is with zeroes on this boundary. And it should work, yeah. It should work. Any comments on other people. Did you get reasonable results, or? Tell me something. Because you guys looked at those graphs and I have not. Any feedback yet? On those? I'm happy to get email, too, about. So all the email, first of all they've corrected the typos in the original coordinate positions. And now they've pointed out I'd better look at M is very, very welcome. It doesn't mean that everybody has to do this, if you've completed that MATLAB assignment, you never want to see it again, and you've kept M=8, it's ok. But if you're interested to see what happens if M goes to 16 or 32, I'm interested also. Right, yeah. OK, so anyway that's the problem we're really thinking about. And that's the problem that is equally important, but it seemed reasonable just to do one of the two. And we were set up to do, we have the code for the stiffness matrix, we would need a new code to do these integrals.

Because this will be linear times linear, right? I'll have to compute that one times this one and I would need new formulas that are not there. I'd need formulas for, this will be linear times linear so I'll be integrating x squared type stuff. And xy's, because I'm 2-D, and y squareds. So it would take a little more code, but not much. I think the math, oh here's a question for you. Here's a question for you. Suppose I have my trial functions, phi_i(x). What do they add up to? Let me again draw a mesh, so I've got a mesh. These are you know, I'm sorry, I want to put in some more triangles here. Lots of triangles, whatever. Let me get some more vertices, too. I'm getting in trouble. OK, whatever. So phi_i, is the piecewise linear guy that is one at node i. So I've got all these different nodes. I need a node there, so I've got one, two, three, there's a node, there's more nodes. If I add them all up, this is just like in an insights question. I've got all these, you could add up these hats in 1-D. What's the sum of the hats in one dimension? One. Good. The sum is one. It's a nice fact that these guys add up to one. And now why is it still true here in 2-D, that these little pyramids will add to one? That's an inside question, but it's worth thinking about. Why do those pyramids add to one? Let me leave that question. I'm thinking about, we haven't imposed any boundary conditions yet. We've got them all. and I claim that if we add up all the pyramids including the boundary chopped off pyramids from the boundary, that we'll get one throughout the whole, now it'll be phi(x,y). Because now I'm moving to 2-D, with pyramids. I think we'll still have one. Let me give you a minute to think about that one. And then we could turn to Fourier questions if you would like, we could do some problems from the text.

Any thoughts about this guy? Why should all those individual pyramids add up to a flat group? Why did it work here? Well, it worked because you could see it, right, somehow? Does it still work if the nodes are not equally spaced? So we've got a hat

function for that guy, and a hat function for this guy, and a hat function for this guy. And these guys are in there, too. We haven't imposed anything. So those one, two, three, four, five functions, five phis, they add up to one and y. Well, you're going to say it's obvious, but that's what professors are allowed to say. Things are obvious, you have to actually say why. Which is not as easy. So, why do they add to one? Let me look inside one element. Why does the sum of these two guys add to a flat top inside that interval?

AUDIENCE: [INAUDIBLE]

PROFESSOR STRANG: At the end points, you've got it. Because what's happening at the end points? This guy, one of the guys, the right guy is one. And all other guys are zero, right. And this guy is also at one. Because it's the right guy. It has height one and all others zero. So at the nodes we are at one, because of one person, really, one element. And then?

AUDIENCE: [INAUDIBLE]

PROFESSOR STRANG: Right. But the sum of them is, why is the sum of them always one, why is slope zero? Yeah. The slopes cancel, right. We know that in between it will be a linear function. That would be one way to look at it. If I add up a linear function and a linear function the sum is a linear function. So I'm getting a linear function, which is one at those points, so what is that function? One. Right, you know that's the straight line. So, that idea will work here too. Look inside some little triangle here. OK, that's got one, two, three corners, OK. And if I look at this sum, what is it at this point? If I look at that sum at this corner, one guy is one, the one for that pyramid. And all others are? Zero. So the sum is one there, the sum is one there, the sum is one there, so that blowing up this little triangle, this is at height one, this is at height one, this is at height one, so what's the roof? Flat. It's just a nice way to see the nice property of these phis. That there's a phi for every node, and they add to one. To that's it. OK, well I was going to say one more thing and I am, about this eigenvalue problem, just because I'll never have a chance again. So this is the moment to say something about the eigenvalues. Lambda. Eigenvalue. I'm answering the question where does K come from, where does M come from? Well, eigenvalue is, boy we really got dramatic music here. That's a great Gates of Kiev, I think might be. Mussorgski. If you like drums and big noise, it's not music actually, but you got a lot of noise out of it. Well, of course, he'd know more than we do, but still.

OK, so the eigenvalues in the matrix case for $Kx=\lambda*M*x$, the eigenvalue problem, lambda, the lowest eigenvalue, lambda lowest, has a nice property. It's the minimum of sort of our energy over our other energy. I just think, well this is something you should see. This is a quotient here. It's got a name called the Rayleigh quotient. And it would appear in the book. So really, I guess what I'm doing is calling your attention to something that's in the book. That this a ratio of x transpose K x to x transpose M x, if I look over all vectors x, the lowest one is the eigenvector. The best x is the eigenvector and the ratio is the eigenvalue. This is like my point that I wanted to mention the Rayleigh quotient. Here it is in the matrix case, and there would be similar Rayleigh quotient in the continuous case. I'll just leave it at that. That in describing eigenvalues, we can talk about $Kx=\lambda*M*x$, like this. Or we can get energy into it. And you remember the whole point about finite elements is, look at the energy. Look at that the quadratics. Multiply things by things. It came from the weak form, it didn't come from the strong form. In the

differential equation here, we just have single terms. We got to these things through that process of multiplying by u's and integrating. That's what gave us these products and it works also in the matrix case. OK, that was a lot of speechmaking about topics that we simply didn't have time for in class. I'm ready for any question, or I'm ready to maybe do a Fourier example, would you like that? Because this is where we really are. I'll even take one that will be on the homework. Just so you'll have a start.

OK, let me take a square pulse, yeah this is a good one, I think. In Section 4.1, there's a question for the Fourier series of a square pulse. OK, so what does the square pulse look like? Here's minus pi to pi. Here's zero. The square pulse goes along here, up square pulse and down. Actually, let me go to L/2, oh I'll just call it h. Let me find the Fourier series for this function. It goes along at 0, it jumps up to 1 over a interval of length 2 h, going from minus h to h, and then back down to and then repeat. So bip bip bip, square pulse. So that's my function. Is that function odd, or even, or neither one? It's even, so I can call that C(x). And figure that I'm going to use cosines for that one, right? So tell me a formula for the coefficients, what's the integral that I have to do? So my C(x) o is going to be some a_0, we have to think what's a_0, then a_1*cos(x), a_2*cos, and so on. So on. a_k*cos(kx). OK, what's the formula for a_k? Before I plug in that function I would like to get the formula. So I'm looking for the formula. It's a formula to remember. So I'm not wasting your time. Because you're going to see it on the board and it'll just take a mental photograph of it. What do you think it's going to be? How am I going to get it? I'll multiply both sides of the equation by cos(kx), right? And I'll integrate. So and then when I integrate, the cosines are orthogonal. Just like the sines this morning. All those terms will go, except for this term. When I multiply this by cos(kx), I'll have cos(kx) squared. Here I'll have a cos(kx), and here I'll have a whole lot of cos(kx)'s but when I integrate, all this stuff is going to disappear. And this will all disappear.

This is it. So a_k is going to be the integral of my function, times cos(kx)dx. Divided by what? Divided by the integral of cos(kx) squared. Because I haven't normalized things. So I don't know that that's one, and in fact it isn't one. So I have to remember to put that number in. OK, so that's the formula and that number turns out to be pi, again. If I'm integrating from minus pi to pi, then the average value of the cosine squared is a 1/2, it's sort of as much above 1/2 as it is below 1/2, and so the average of the half, the interval is 2pi, so pi. OK, that's the formula. Please just take a mental photograph. Catch that one. Alright, now I've got my particular C(x), my square wave, square pulse. Very, very important. Very important Fourier series here. Famous one. OK, so what do I have? From minus pi to pi, so what's my integral? Well, my integral really doesn't go from minus pi to pi because my function is mostly zero. Where does my integral go? Negative h to h, right? And in that region, what is C(x)? One. So you see it's going to be nice. From negative h to h, where this is one, I just have to integrate cos(kx), so what do I get? sin(kx), over k, and the pi so you see again that that k is showing up in the denominator, and that's going to give me the typical decay rate of 1/k for functions with steps. For step functions. And now I have to evaluate this between minus pi and pi. And no, h. Better be h. I mean, minus h and h. So what do I get for that? I get sine(kh), right? At the top, and what do I get at minus? So I now I want to subtract, what is the sin(-kh)? It's a negative, right? So as I expect with an even function like cosine, am I just getting twice? I could take it from to h, and it would give me one of them and the other one. Yep, I think so, and divide by k pi. So those are the Fourier coefficients. Except for a_0. a_0 has a slightly different formula, because for a_0, why is a_0

different? How do you come up with a_0, and what's its meaning? a_0 has a nice meeting, so this is worth having come this afternoon for.

a_0 will be what? Well, I could get it the same way. What will I multiply both sides by? If I want to pick off a_0? Just one. It's not a cosine, it's the $\cos(0x)$, it's the one. And then I integrate. I'm just going to get the integral from minus pi to pi of $C(x)$ times one, divided by the integral from minus pi of one times one. Same method. multiply both sides by one, which was the very first of my orthogonal functions. Integrate it, all the other integrals went away, right? The integral of cosine over a whole interval. Its periodic. You get the same at the two ends, so the difference is zero. So we just, the only term left was a constant. And now what is the integral, row what's the denominator now? That was the little, slight twist. 2pi. The denominator's 2pi. Yeah. That's that's why it's not, yeah, it's slightly irregular, I have to divide by 2pi. And now, what word would you use to describe, if I have a function, and integrate it, and I divide by the length, what am I getting? There's an English word that would describe what this is. Average. This is the average. And it has to be. This constant term is always the average. And what will it be for this? So this was a_k, and what is a_0, then? So you can now tell me, so everybody's remembering this formula, you integrate the function and divide by the 2pi. Now we've got a particular function, so what is the integral of that function? So what does this equal? For this particular $C(x)$? What's the area under that function $C(x)$? 2h. Right? So 2h/2pi cancel twos. So there's a constant term, a_0 is h/pi and the and the cosine terms are, yeah, actually we're going to get something nice. A really nice way to complete this will be if I put this together, put this series together. So now I'm saying that this square pulse is that constant term h/pi plus the next term a_1, you can see all these terms have 2/pi's.

I'm a little surprised that h over, yeah, I guess it's right. 2/pi. So I've got sin(h), I think. And now I'm just copying this. 2/pi*sin(h), sin(h), is that what I want? Over one. That's the coefficient version of sine, of $\cos(x)$. a_1 was the coefficient of $\cos(1x)$. And then a_2 is the coefficient of $\cos(2x)$. So that will be sin(2h). k is two, so I have a two down here, $\cos(2x)$. And so on. Yeah, I think that's the Fourier series. That would be the Fourier series for the square pulse. Yeah. That would be the Fourier series for the square pulse. Could I test any interesting cases? Suppose h is all the way out to pi. Suppose I take that case. Let h go all the way out to pi, then what's my function? If h=pi, then what have I got a graph of? Just one. It's just a one. If h is pi, what happens? That becomes a one, and what about these other things? What is this thing when h is pi? Zero. All the other terms go away. It's just a sin(2pi) that would go away. Yeah, so if h is pi, if I go out to the place where I don't have any jumps at all because it's now all the way out there, then these terms all disappear and I just have this.

And I would like to ask you and it's going to come up on Friday, too, what happens if h goes to zero? Well, let me just take h going to zero. What happens to this whole thing? What happens to my function if h goes to zero? Goes to zero, right, then squeezed it to nothing. And if h is zero then sin(h) is zero, I get 0=0, that's not interesting enough to mention on Friday But there is one case that is important. Suppose I make the height, yeah. Make a guess. Suppose I make the height higher as I make the base smaller. I'm going to keep the area as one, so if this has a base of 2h, I'm going to have a height of 1/2h. So if I keep the area at one, so the height now is 1/2h, so now my square pulse I've divided it by 2h. I have a 1/2h multiplying everything. And now if I let h go to zero, something more interesting will happen. And what? Just tell me first, what would you expect to happen? Delta. Right, delta.

So what I'll see show up will be the Fourier series for the delta function. When I divide by 2h, so I have sin(h)'s over h's, and of course what's the great fact about sin(h)/h? As h goes to zero, it goes to, everybody know, that's the big deal. Yeah. One. sin(h) is the same size as h for a very small h, and approaches one. Yeah so we'll see the delta function Friday. OK, so you've got a sort of mini-lecture instead of a real chance to ask about homework. Next Wednesday should be different because there will be Fourier series homework, and I'll be ready to answer questions about it. OK, thanks.