

**PROFESSOR:** All right. Welcome back to 6849. You may recall or have heard that this film crew from Japan is here. They're from a program called [? Gachan, ?] which is about showing all the cool things that happen in universities around the world to students in Japan. So don't mind them, but you're going to have to sign a release form at the end. I apparently don't need you to sign a release form. It just works.

We're talking about origami today. We began a series of lectures on computational origami. We're going to do some design, some actual folding, and some foldability algorithm stuff. Does everyone have one of these strips and a square of paper? You don't. Who has the master? Why don't you grab a few? Some of these. Here's the package. Lots of stuff.

The paper comes from the [? Gachan ?] team, so thanks to them. This is stuff you use in your adding machine, I guess. That's where it comes from. Yes. You need one of each. Go for it. Open it. Let's get started. Where is my chalk?

So just to warm you up, I want to do some folding and give you some terminology. I know some of you have done origami before, but a lot of you haven't, so bear with me. I will try to do this quickly. And even for those who have done origami, you may not have done origami math before. That's why you're here.

So things like a piece of paper, which you might take for granted, I'm going to define here to be a 2D polygon. There's actually many interesting definitions of a piece of paper, but for today at least, most of the time, we'll think about polygon. Usually, you think of a square piece of paper, but "polygon" would be the geometer's generalization of that.

Also, the polygon has distinguished sides. So distinguished top and bottom side. In practice, with standard kami paper from Japan, you have the colored side and the white side, so that's one clear distinction. Maybe we call the colored side the top. Doesn't really matter, just so we can tell things apart, and in particular tell mountains from valleys. That's our motivation.

So a crease in a piece of paper, this I'm going to think abstractly as just a line segment or some one dimensional curve drawn on the piece of paper. And then we have crease pattern. This is just a bunch of creases, so a bunch of line segments drawn on the piece of paper. If you are graph theoretically inclined, I would think of this as a planar graph drawn on the paper. If you're not graph theoretically inclined, just ignore that definition. They're the same thing. That's a crease pattern.

So for example, the thing we're going to be folding is this very simple pinwheel, and this is an example of a crease pattern that folds into that pinwheel. So here, the lines are distinguished. There's two different kinds, the red lines and the blue lines, but ignore that for now. That's a bunch of creases which you could put in and make that thing.

Before we get there, I want the notion of a folded state. So this is all pretty normal if you're an origamist. This is really something only the mathematicians use, this term, folded state. Formally, this is what we refer to as the finished product. This pinwheel is a folded state of this piece of paper. Clear?

Defining that is a real pain, and you can look at chapter 11, I think, in the textbook. Maybe I'll talk about it at some point, but I don't want to get distracted by how you define what a folded state is. But you're not allowed to stretch, you're not allowed to tear. It should be a nice, valid folding. You can't cross yourself. So that's this notion of folded state.

And if you take a folded state and then unfold it-- this is something you can do with a folded state-- you get a crease pattern. So I can take this pinwheel, unfold it, and I can see which lines are creased to make that pinwheel happen. So that's the crease pattern of a folded state.

Today especially, and a lot of the field is concerned with the idea of flat foldings, which are a particular kind of folded state that lives in the plane, lying in the plane. Of course, it doesn't literally lie on the plane. There's lots of layers stacked on top of each other, but geometrically, everything has been collapsed onto one slightly

thickened plane here. So this is a flat origami. This is also a flat origami, the trivial kind. That's a flat folding. And when this is possible, we call that crease pattern flat foldable.

And finally, we get to the idea of mountains and valleys. So a mountain crease is when the bottom sides touch, and a valley crease is when the top sides touch. So remember my distinguished sides. I'll call the colored side the top, so then this is a mountain crease over here because I'm bringing the two white sides together, and where's a valley? I guess this is a valley, where I'm bringing the two colored sides, the two top sides, together. Together, that makes this pattern, and hopefully, that matches up with here.

I mean, it's symmetric. You could flip everything over. You'd just get the inverted color pattern. It doesn't really matter which side you defined to be which, but the point is to distinguish which are mountains versus which are valleys. You get different parity, it looks like, I have matched up there.

So let's fold this, just for some experience, if you've never made one before. Take your square piece of paper, and to get started, let's say with the white side up, just fold along the two diagonals, the valley folds, bringing the two white sides together. So you fold along one diagonal, then you unfold, then you fold along the other diagonal. How are we doing? Couple diagonals. I have the big sheet. I should be going slower.

So now, I want to fold that inner square, and we've conveniently marked the center point. So I'm going to fold each of the four edges of the square to the center, and make sure to line up the left and right edges, and that guarantees that I fold parallel to that bottom edge. In this case, I just want to crease the middle half of that line. So I'm not creasing all the way. I'm just creasing in the middle half here to make that one edge of the square, and then I turn and I repeat that four times. Switch sides for different visibility.

This is not the usual way to teach this model, but I think it's the mathematically appropriate way. I'm trying to make that crease pattern as quickly as I can. I did it

with the white side up because I'm folding everything valley, which is easier. In reality, I want these to be mountains. When I turn them over, they're all mountains, and I'm happy. We made some extra creases here, which we didn't need to.

How are we doing? How many people are done with those? How many people are not? Couple people. Sarah, hurry up.

Now comes the fun part, and this is to give you an idea of what general origami folds can be like. I want to simultaneously fold all those creases that I put in except for the center, where it's supposed to be uncreased. So I'm going to bring those four edges of the square in, and if you do it right, you'll end up with these four flaps sticking out of the top.

It's not going to lie flat. We don't have enough creases in yet. But we have these four flaps of the pinwheel, and then you just get to choose, do I flip them over one way or the other? If you flip them over as much as they go in that direction-- I'll try to match the figure. I'll put this guy over at the left, crease him in, put this guy down at that end, and this guy over to the right, and this guy up. I get my pinwheel. If that doesn't work out for you, you can try again.

It's not so hard, but it gives you some idea that putting all these creases in at once is kind of necessary for this to happen. In particular, if you look at the crease pattern of this model, there is no fold that goes all the way through the square, so you can't make a first fold and then make a second fold and so on. You really have to do this all at once. There's many ways to do it, some of which are more intuitive, but I thought this simultaneous collapse is the coolest. It's nice and fourfold symmetric as you do it, I guess, if you do it right. I think this even folds rigidly.

That's to give you some intuition, some real, practical origami experience. And you get an idea of what a crease pattern is, when a mountain valley is. I have two more terms and then we can go to math, some real stuff.

There's two notions that involve mountains and valleys. One is a mountain-valley assignment, and this is something that goes together with the crease pattern. And

this specifies which creases are mountains and which are valleys. Did you all get it? Good. The pinwheel, I mean.

So you could think of this as a coloring of the edges, whatever. In this case, I'm simultaneously using color red to denote mountains and blue to denote valleys, and also, the standard in origami world, although it's a bit annoying, is dot dash means mountain and dash means valley. That's pretty common. You'll see that throughout. That was invented in the 1950s, I guess, by Yoshizawa, maybe earlier. Somewhere around there.

And then if I take a crease pattern together with a mountain-valley assignment, I'm going to call this a mountain-valley pattern. That's the picture that I'm really drawing here. Sometimes it's useful to think of these as coming together, sometimes as separate. You're going to see this terminology all over the place, so I figure I should define it. And sometimes, m is mountain, v is valley. Now we can do some fun stuff.

I showed you this pinwheel to give you an idea of more complicated folds, the general origami folds. But today, I want to focus on relatively simple kind of folds, which is called simple folds. Unlike this thing where I have to simultaneously collapse many creases, a simple fold just involves one crease. It is a fold along a single line, and we fold it by plus or minus 180 degrees.

So depending which way you're counting, plus 180 is maybe valley, minus 180 is mountain. If I start with a flat folding, after I make a simple fold, I will again have a flat folding. What we're interested in is, what can you make by a sequence of simple folds? I start from a square paper, say, and I make a sequence of simple folds. At all times, I will be flat folding. It's easy to get there.

The pinwheel is something you could not make in this model. It's a restricted model, but it's still surprisingly powerful. You still get a universality result. Even with simple folds, you can make any flat thing. And if you relax this constraint and don't say it's plus or minus 180, then you can actually make any 3D thing you want.

So that's the first thing we will prove today, and that's in the origami design world.

And then we'll go to origami foldability. I'll give you a crease pattern. I want to know, can it be folded by a sequence of simple folds? That's a much harder problem, but some cases are easy.

So this the universality result I mentioned in lecture one. This is going to be the inefficient version, but we're going to prove that everything can be folded from a square paper. I'm going to be more precise about what I mean by everything. So I'm going to imagine there are a bunch of polygons in 3D and they're somehow joined together into one connected mass. This could be a polyhedron like a cube, it could be a polyhedron like a bunny, it could be some crazy, thorny mess, but each piece locally is a polygon, and you just stick a whole bunch of them together to some connected thing.

Plus you get to specify what color, what side of the paper you want to be visible on each of those polygons. You can even do that on each side of the polygon if you want. And you can fold that thing from a big enough piece of paper, such as a big enough square, a big enough rectangle, whatever you feel like.

Bicolor paper here refers to the idea that your paper is different colors on one side and the other, so white on one side, colored on the other side, white on one side, black on the other side, whatever. We usually try to draw that with the folded over corner. There we go, bicolor paper.

So for example, for fun, here's how you could imagine this design by John Montroll. This is folded from a square of paper, white on one side, black on the other. It's a zebra. And you could imagine the input to that problem was a bunch of polygons. There's some white polygons, some black polygons. Together, they form a connected union, and that could be the input. This could be an output to how you fold your square to make that color pattern. Of course, if you gave this input to that algorithm, you'll get a much uglier folding than the one designed by a human, but that's the idea.

You could also imagine you're just given a polygon you want to make. You say, I want to make a horse, a flat horse. Then you could give that to this algorithm and it

will tell you how to fold a horse. That's the general idea. So how do we prove that? Want to make everything.

So this is an early result in computational origami design, I think from 1998, probably. Trial version here is in 2000. Any ideas how to prove this? How can we make anything? Got to get you thinking a little bit.

We have a subtle hint. So the idea is to take a really long strip, much longer than this one, and fold it, and just wrap around your polyhedron over and over and over, and eventually make whatever you want. So it's going to be a little flimsy, but mathematically, it'll work perfectly.

So first thing to do is fold your paper, whatever shape it is, down to a long, narrow strip, so a long, narrow rectangle. Conveniently, you have your strips of paper already, but if you started with a square, the first thing you'd do is accordion pleat it down to just one edge of the square. Crazy already. If you started with a square, this is a bad idea. If you started with a strip, it's actually pretty good. It's probably close to optimal.

But now, how do I actually, say, wrap everything with the strip? Let's be more formal about that. The first thing I'm going to do, which is a common trick in computational geometry, it's like, well, polygons are confusing. I like triangles better. They're much simpler. So just triangulate all the polygons.

If I have some crazy polygon, imagine that's some piece of the zebra, I'll just subdivide it into triangles. There's standard algorithms to do this. I don't want to go into that. You can just add a bunch of these diagonals to the polygon, just keep adding them as long as they don't cross each other. When you're done, it will be triangulated. You can do that efficiently, linear time, even.

That's a little bit easier to think about. Now I just have to figure out how to wrap one triangle and then how to combine those wrappings together somehow. Then I could go through triangle by triangle, visit everything, cover everything I need to cover. So that's what we're going to do.

Cover each of these triangles in some order. We're going to cover each one at least once, possibly more if we're not careful. For starters, we'll be not careful. And we're going to do this by using a zigzag path parallel to the next edge starting from the opposite corner. So some technical details. Much easier to see in a picture, so let me draw you a picture.

Let's say we want to visit this triangle first, and after we visit this triangle, let's say we've planned out that we're going to visit this triangle next. After that one, we're going to go to this triangle. So this is what I'd call the next edge, because it's the edge connecting to the next triangle. So I'd like to zigzag parallel to this edge, so I want to go back and forth like that.

And then it says I want to start at the opposite corner, which is this one. So I'm going to start here, and I'll draw this a little more precisely. I'm going to take my strip and just go this way for a little while, and turn around and come back this way, and then turn around, and we're going to see how to do turning around in a moment, so something like that.

Now I've got to start planning a little bit, because I want to do this triangle in the same way, meaning I want to go back and forth parallel to the next edge starting at the opposite corner. So that means next thing I want to do is start at this corner, so I need to set this up so that I don't end here. I'd like to end here. You can do that just by doing one more iteration.

If I end up with the wrong parity-- here it looks like I won't too much. Here it's a little ugly. Got a little bit of uncovered portion of the triangle, so you need a slightly different turn here. Then I'll just come and sort of overlap myself. That's OK. I just have to cover it at least once. And then I do some kind of turn here, and then I'm going to do this.

So there's obviously some questions here. How do I make these 180 degree turns? How do I make an arbitrary turn? How do I turn even when I'm allowed to overlap myself? I'm not going to worry about that too much, basically the same. Basically, how do I turn? How do I turn?



Well, let's take our strips and do some turning, and I have a slide to follow along. I have a big strip, a little bit too big. So the idea is you have your strip, you're gone merrily along, and at some point you decide, oh, I really need to make a 132 degree turn. I mean, it could be anything because here, we've got to make some crazy turn. We don't know what the amount is.

So the turn gadget is always the same. You start out by making a mountain fold perpendicular to the strip. So just fold in half, so to speak. Of course, half of this is connected to everything you built already. The other half, you don't know. And then you take the back layer and fold it to whatever angle you want. You have this degree of freedom wherever you want to put it. So for example, if we wanted to make an obtuse angle.

I'm always going to fold incident to that bottom corner. It's a little hard to handle, but you get the idea. That's a nice, perfect turn right there. Looks even cooler on my side. You just see the angular bisector or something. So that's a decent obtuse angle.

Things are little bit uglier when you make an acute angle, something like this. You could make whatever angle you want. You should have enough strip to make several turns if you feel like it. If I make an acute angle like this, I get some overhang. So I have what I want, which is the strip coming here and the strip coming here at the extremes, but over in this corner I have some extra material, and it could be a lot.

But whatever it is, I'll just rid of it by folding it underneath repeatedly. In this case, one fold, it's gone. In general, when I make that fold, it might come out over here and I have to wrap around that cone several times. There's probably an example of that here.

This figure in the middle version, you start just the same by folding a super acute angle. I fold that corner over and it overhangs a little bit on the bottom, but it got tinier. And if you just wrap it around the corner, eventually it will disappear and you

have exactly what you want, which is a strip coming horizontally and then a strip coming off diagonally at whatever angle you wanted. So that's really easy. That's what we call a turn gadget.

So we do this using turn gadgets. Now, you can't actually turn with 180 degree angle here, which is why in this picture, what I drew was a 90 degree turn and then another 90 degree turn. You can do that however you want. That's one way.

In fact, the way I've shown it, it would be orthogonal, but doesn't really matter. These turns are not much harder. You just do one or two of those turns to get aligned. And you can decide here, do I turn around to the left or do I turn around to the right, and then I zigzag? Depending on which way I go, I will end up either at this corner or this corner, and if my next triangle's up here, then I want to end up at this corner. If my next triangle's over here, then I want to end up at this corner.

You plan out ahead of time what order you're going to visit the triangles, just decide. Technically, it's like the traveling salesman problem, but you don't have to be efficient at the moment. We're not worrying about efficiency. Just visit triangle after triangle, keep going to one you haven't visited before. It could be far away from you. Maybe you have to go over many triangles you visited before, but eventually you'll get to one you haven't visited until you're done, and then boom, you've made anything. Amazing.

We haven't quite proved the theorem. At this point, we can make any connected union of polygons. This is one polygon, but if you're in 3D, when you cross over from one polygon to the next, you might have to bend it at some angle that's not flat, but then you can make any 3D thing, too. Yeah?

**AUDIENCE:** When you wrap around the triangle whose edge type is the bottom, how do you do that when the top left triangle is there?

**PROFESSOR:** Sorry. When I wrap around here?

**AUDIENCE:** No, the triangle whose edge is the bottom edge.

**PROFESSOR:** This one?

**AUDIENCE:** Yeah. How do you do that, given that there's the triangle in the top left?

**PROFESSOR:** Oh, I see. The question is, what do I do here because I'm going to overlap into this triangle again?

**AUDIENCE:** Right.

**PROFESSOR:** Yeah. So you have to be a little bit careful here, and there's one other gadget I need to mention here, which is the hide gadget, I think it's called. Because I didn't fold exactly this triangle, I folded too much. At this point when I have the strip here, and maybe a little bit out of the way, I'd like to get rid of all this messy stuff, just tuck it underneath.

So I'm going to fold along this line, fold all this stuff back behind, and that might still be a mess because it could stick out over here or somewhere else. So let me tell you about the hide gadget a little bit, maybe over here. You have to be a little careful when you do that exactly, but it's not that hard.

So in general, the hide gadget is you have some convex polygon. Here, it's always going to be a triangle, actually, but this would work for any convex polygon. You have some folding which contains that polygon, but it might go outside all over the place. That's our situation. We want this triangle. We have a folding that includes the triangle but it has all this junk on the outside.

All you need to do is repeatedly fold along the extension of each of these edges, and you can prove-- I'm not going to do it here because it's a little bit messy-- but as you fold that thing, obviously the area that you go outside gets smaller, and you do make steady progress and eventually, after finitely many folds, you will have reduced this thing down to the convex polygon. Just keep folding mountain fold along each of those edges. And I think there's even a pseudo polynomial bound on the number of folds you need to make, although I won't go into that here.

So that's what you do with this triangle. Then you fold the next one, and then you

apply the hide gadget to that triangle. When you do that, there is this triangle in the way. Technically, I guess, that wouldn't quite be simple folds because you're folding along this line. You need to tuck it in between the layers. You have to avoid collision with this triangle and go underneath this one. Hm. I hadn't thought about that.

I'd like to say this works with simple folds. It sort of does in that each fold you make, you're just folding along one line at a time, but you do actually collide in the middle. It's not so ideal. Origami folds, this definitely works. I wonder if there's a way to do it with simple folds now. Interesting. Hm. I don't know.

**AUDIENCE:** Can you undo the corner gadget to separate the triangles, hide the folds, and then put them back?

**PROFESSOR:** The question is, can I pull them apart, do them separately, and push them back together? Pulling them apart and pushing them back together would not be simple folds, I think. Maybe you could do it.

**AUDIENCE:** You just type a [INAUDIBLE].

**PROFESSOR:** Yeah, OK. Interesting. We should think about that. I'm surprised this is not known, but I'm pretty sure now that it's not known. Maybe first open problem session, we can solve that in a few minutes. Should be doable.

There's one more gadget we need if we want to get the two color pattern. This polygon maybe is all one color. Maybe it's all the colored side of the paper. But then at some point, we might transition over to another polygon, and maybe that's the other color.

So all we're going to do is we're, again, visiting these triangles in some order. Whenever we go between a black triangle and a white triangle, we are going to apply a color reversal gadget, which is very simple. I forgot to color one side of the strip, so bear with me for a second, and feel free to color in your own strip. Probably killing this microphone right here. Purple on one side, white on the other, although it's so transparent, it looks almost the same.

The idea is very simple. Where are we starting from? We're starting from the not purple side. Then we make a mountain fold. It's just like a turn gadget. In fact, it's just like you're turning 90 degrees. We fold up like this, and then we just immediately make another 90 degree turn without the first fold, and we get a sharp transition straight along the strip from light purple to dark purple, it looks like, but from the white side to the purple side. Right when you transition between any pair of triangles of the opposite color, you just apply this gadget. You make three more folds, and boom, you've got your color reversal.

**AUDIENCE:** Can you do that again?

**PROFESSOR:** Do it again? All right. So first I make a mountain fold. Now both sides are white. Then I fold this guy up, so now I have a regular 90 degree turn, and then I fold this guy back. That's the tricky part because you can't see what's happening. Let me flip it over for you. So I'm just folding the top two layers here along that 45 degree angle, and then I get that transition. On this side, it's still purple and white but it's a little uglier.

It's on the slide. You can try again. It doesn't say here, but fold just the back layers. All right? Questions? So that's how you can visit all these polygons, do color transitions, you can do it in 3D, make any polyhedron. A little crazy, but the point is you do need to do some algorithmic construction to really see this method work. You need those three gadgets. You need to be sure that you can actually combine them in the right way. And I'm hand waving a little bit some of these details. You should be more careful, but it works.

It's obviously horribly inefficient, but we can make it at least a little bit more efficient. Let me tell you about that. So we can achieve something I call pseudo efficiency, which is if you are allowed to start with any rectangle you want, so in particular, if we start with a long strip of paper like you did here, then you can make this super efficient.

The area of the piece of paper that we start with can be equal to the surface area of the thing that we want to fold. If we could achieve this, we would be really happy.

That means we have zero wastage. I can't get zero wastage, but I can get just a little tiny bit of wastage. So we write epsilon for a very tiny result, very tiny error. So epsilon does have to be positive, we have to have some wastage, but it can be arbitrarily small. Basically, the narrower and longer you make your strip, the closer the area of your piece of paper gets to the surface area of your target shape, which is the best you can hope for.

So with this crazy proviso, which is completely impractical we get a super practical outcome. And the summation is not very practical, but it's kind of nifty, and I particularly like the proof technique. It uses a very powerful tool in computational geometry, so let me show it to you.

So the central issue is that while visiting a triangle, we actually do pretty efficiently. Except for the very last strip, where we might overlap ourselves, except for these little corner turn gadgets, and also the material that's used up in the gadgets themselves, We're actually really efficient. We're covering this triangle almost one for one. There's a little bit of garbage at the edge and a little bit of garbage at this edge, but you could imagine as the strip gets smaller, this wastage gets smaller, because it's essentially this length times the width of the strip up to constant factors. So as the strip gets narrower, it gets closer and closer to this thing times 0, so this wastage gets closer and closer to 0. Same thing over here, same thing over here.

The real problem at this point is that we visit some triangles possibly many times. I was totally free about how I visited triangles. Maybe I go here and then over here and then back over here. I may be very wasteful. So I just want to avoid that wastage. Just visit each triangle once, exactly once.

How could you do it exactly once? With a great idea called Hamiltonian refinement. I've used this idea many times. This was the first time I used it. It's very cool. I think it goes back to '97 or something.

So you have some triangulation. I'm going to keep it relatively simple. And I'd really like to visit each triangle exactly once. So you think, OK, maybe I'll start here, then I'll go over here, and over here, and uh oh. Then I'm stuck. I can't get to the other

triangles without revisiting an existing triangle. It's impossible to find what we call a Hamiltonian path. Hamiltonian path will visit each triangle exactly once.

But who cares about these triangles? I just made these triangles to make it easier to think about a polygon. I could actually cut these triangles into smaller triangles and ideally make it Hamiltonian. So here's the idea. There's this idea called the dual graph, which is also pretty central in computational geometry, where I'm going to make a little dot for every triangle inside each triangle, and then I'm going to connect those dots whenever there is two triangles that share an edge.

This is called the dual graph. These parts are called the dual graph of the triangulation. And in this picture, I've got a nice, kind of tree-shaped graph. In general, you might get cycles, but then just throw away edges until you get a tree. I really like to think of it as a tree, and if I throw in enough edges, it will be a tree.

Then my idea is I'd really like to just visit everything in this order, walking around the tree. I can go around in a cycle and visit each of these little polygons exactly once. Now, these polygons aren't triangles, which is kind of annoying, so I just triangulate. So you could just triangulate like this, I guess.

If you want to be excessive and simpler, from every one of these dots, you can cut to the midpoint of the three edges and also cut to the vertices. For every triangle, I just cut to the three vertices and cut to the midpoints. You do that, and you are guaranteed. This gets a little messy, but this is theoretical.

You will increase the number of triangles by a factor of six. Each triangle gets replaced by six little smaller triangles. It covers the same region, and yet now it's Hamiltonian. Now I can visit these triangles and visit each triangle exactly once, and that's how you get pseudo efficiency. Pretty easy once you know this trick.

Questions?

**AUDIENCE:** Why is it called pseudo efficiency?

**PROFESSOR:** Well, I made that up. Why is it called pseudo efficiency? Because it's not really efficient. It is efficient, I agree, but only with this crazy assumption that you're

allowed to start with an arbitrarily narrow strip, and that's not really practical. The practical origami problem is I start with a square. How big of a square do I need to fold this shape? And that's something we'll talk about probably next week. That's been studied but it's a lot harder, so I'm starting with the easy stuff.

**AUDIENCE:** Does this necessarily keep the mountains and valleys that you defined at the beginning?

**PROFESSOR:** Will this keep the mountains and valleys the same? The input didn't specify where the mountains valleys should be. It just said you have to make these polygons. Every fold you make in this construction will stay, but you may make more folds that go on top of existing folds. I think all the mountains and valleys that you define here will remain mountains or valleys as we specified, but keep adding creases until you finish. Yeah?

**AUDIENCE:** Are you just doing a 2D case of what we call the 3D pyramid there? [INAUDIBLE]

**PROFESSOR:** The question is, what about 3D? This dual graph idea does work in 3D. I drew a 2D picture because it's a lot easier to draw, but you could imagine, for example, you have a cube. I'm not going to be able to draw everything, but you can still form a dual graph.

I should have triangulated the cube, but in fact, you can do this for any kind of graph. I'm making a dot for every face of the thing that I want to make. Each of these polygons, I make a dot. So in fact, if I finished this picture, I would get the dual graph is the octahedron. It would look something like this topologically.

It doesn't have to be a geometric thing. It works just as well in 3D. It actually still works even if you just say, for every triangle, just cut to the midpoints of the three edges and to the vertices. Then that thing will be Hamiltonian and you can walk around it by doing this dual and spanning tree construction. I won't go into it exactly, but that could be done in linear time.

I'm still only making a two dimensional surface. I'm not trying to make a 3D solid here, so that's why locally, it's two dimensional. Locally, this just works. Another



question?

**AUDIENCE:** So it seems that if you could also just stick with the original construction, and when you cannot find the Hamiltonian path around the triangles, [INAUDIBLE], you don't cover them again, you just get a narrow strip that runs through them and just rigidly [INAUDIBLE]?

**PROFESSOR:** That's a neat idea. So instead of doing this Hamiltonian refinement, you could visit the triangles in any order, but then to get from one triangle to some distant triangle, instead of covering all the triangles in the middle which you've already covered, which is excessive, just go as straight there as possible. Locally, you'd think of that as being a one dimensional wastage instead of a two dimensional wastage. So probably you can get the same result by that technique. I hadn't thought of that before. We should check it, but I think that should work. This is kind of cooler, just because it uses a fancy tool, but I imagine with a little more analysis, you could prove that that works because you know there aren't too many triangles. Good.

I think I'm going to skip seem placement and move on to foldability. This was a little intro to origami design. Obviously not practical yet, but it's something. It's nice and universal. You can make everything.

So let me tell you a little bit about the other side of the world, which is, instead of a target shape, what if I gave you a crease pattern? We just want to know, does this crease pattern fold? I give you the crease pattern of that pinwheel. I say, does this fold into anything? Does it fold into a flat origami? Does it fold into something else? Can you fold it with simple folds? Can you fold it with origami folds? Those are all interesting questions.

We're going to start with a very simple form of it, which is when the piece of paper is one dimensional, which is almost what you were thinking of here. It's like strip folding, but I'm going to furthermore require that all the folds I make are perpendicular to the strip direction. So I'm always going to fold like this and then like this. So it's going to remain a strip. It'll just be a smaller strip. I know it seems kind of

boring, but it's something we understand really well, so that's why I'm going to tell you about it. And then it relates to locally what happens at one vertex of the crease pattern, but we'll get to that next class.

So one dimensional flat folding. I don't have them anymore here, but I'm going to redefine "piece of paper" and all these things because I want to be even simpler. So piece of paper now is going to be a one dimensional line segment. A crease is going to be a point on that segment. In general, a crease is one dimension smaller than the piece of paper you're working with.

You could actually define "piece of paper" to be a five dimensional solid if you wanted. Then creases would be four dimensional flats, they're called. It's not a hyperplane. Four dimensional flat. That's what they're called. Or if you wanted to do curved creases in five dimensions, then it's a four dimensional surface somewhere in there. We won't do 5D origami in this class.

Flat folding now means that your folding lies on a line. Of course, it will always lie in the plane if you're folding a line segment in the plane. So generally, we take a piece of paper one dimension, we fold it in a space that's one higher dimension, so we're going to be folding this thing in 2D. We use increases that are one lower dimension. Here, zero dimensional points. If we want it to be flat, it should in the end lie in the same dimension the piece of paper did.

It's great because I can just draw. There's my piece of paper. Here are some creases. They're spread out however. Maybe they're marked mountain and valley, and I'm not going to use dashes and dot dashes here because they're points, so I'll write m's and v's, something like that. And I didn't check, but this either will be flat foldable or not. I think it is. How do I know that it is? Because we have a great characterization that tells you when a mountain valley pattern is flat foldable.

Now, the first question you might ask is, what about crease patterns if I don't specify mountains and valleys? Then you can always fold anything. If I just gave you a segment and I put creases in some crazy pattern, all I do is alternate mountain, valley, mountain, valley, mountain, valley, mountain, valley, and I'm going to regret

making such a big example. And I just go mountain, valley, mountain-- it alternates. You get some zigzag. And this thing is not going to collide with itself. It just works. In fact, you could even fold it one step at a time left to right by simple folds, and you'll get a nice flat folding of that thing.

So every crease pattern can be folded with some mountain-valley assignment, namely the alternating accordion pleat, but what if I give you the mountain-valley assignment? Sometimes I can do it, sometimes I can't. So here's an example where you can't, two valleys with some really big segments on either side. So when you try to fold that, it looks something like that, and if one of these was really short, one of them could tuck inside the other, but because they're both long, they have to collide. That's the sort of thing we're worried about and we'd like to detect that efficiently with an algorithm.

So let me tell you how. We're going to use two different operations, and both of these operations will, in fact, be performable using simple folds. What we're going to end up proving is that a one dimensional mountain valley pattern is flat foldable if and only if it's flat foldable via a sequence of simple folds. So these simple folds that we've been arguing about are universally powerful for 1D paper. They are not universally powerful for 2D paper because you cannot fold this pinwheel with simple folds. There's no first fold to make. But in one dimension, turns out there is. It's not obvious, but it's true.

And here, we're going to make two very simple kinds of simple folds. The first one is called an end fold, which is you take the last, or the very first crease, and you just fold it over. That's an  $n$  fold.

Now, for this to be valid in the simple folding world, to not get stuck, I'm going to require that this portion has no creases inside it. So it could have a crease right at the boundary, but there are no creases in that region. So when I fold this thing over, I don't cover anything up. No creases get covered up.

This in some sense is clearly a good thing to do. This may not exist. If you find such a crease and you make this fold, now you can think of this piece of paper as sort of

glued up there. You can think of just you cut off one part of the paper. You made a problem even easier. So this is clearly a safe thing to do. It should make your folding easier, but usually it doesn't exist.

In that case, we make what's called a crimp, which is we have our segment. We take two creases with different mountain-valley assignments. Maybe the first one's mountain, second one's valley, but they're different, and then we fold those two creases. Something like that. So the paper get smaller again, but here it's a little less clear what's happening because there were creases over here, creases over here. They got closer to each other. Maybe that's a problem. Maybe it messes you up. Turns out it doesn't.

The other thing, again, we require, if this is distance  $x$ , we require that this region of length  $x$  and this region of length  $x$  is empty of creases. There's nothing there. You could have things on the boundary but nothing interior to here, nothing interior to here, and nothing interior to here. Just two increases with sort of a nice, safe neighborhood around them so that when I make these two folds, this whole region, I didn't cover anything up in there. There's nothing in this Zorro mark.

That's the definition of an  $n$  fold and a crimp. Now, maybe these things don't exist. I claim if they don't exist, you are not flat foldable no matter what you do. These are enough to make everything that is flat foldable. We're going to prove that.

We're going to characterize flat foldability by saying, if I gave you some mountain-valley pattern, and I want to know, is it flat foldable? Remember, a mountain-valley pattern is a crease pattern together with  $m$ 's and  $v$ 's written on each crease. I claim it is flat foldable if and only if there is a sequence of crimps and end folds. So this is saying that crimps and end folds are all you need.

In fact, we're going to show that you can make any crimp and any end fold, anything that looks valid, anything that has these empty regions here, just greedily keep doing one after the other without regard to what's going to happen in the future. If you get stuck, then your thing in the first place was not flat foldable. If you finish,

obviously, it was flat foldable. You've folded it. So it doesn't matter even what order you do these operations. I won't write that here, but we will prove that.

And there's another characterization which we use to prove this one. This is the one we care about. This one's a little bit easier to think about. I call it the mingling property. It says mountains and valleys hang out with each other, basically. That's why I call it mingling.

So what does it mean? If I take a whole bunch of v's in a row, or I take a whole bunch m's in a row, as many as I can, that means they're surrounded by creases of the other type or possibly the edge of the paper. But if I take a whole bunch of v's, then the next crease and the previous crease are m's, let's say, then the adjacent crease of the other type, or possibly the end of the paper, and I want this to be true on at least one side. This is, again, one of those things that's much easier in a picture, but I will write down the words first. Is nearer than the adjacent m or v. Sorry, backwards. I'm trying to use parallel construction here but I got it inverted for my notes.

Let me draw the picture. You have a segment. Suppose I take some maximal sequence of m's all in a row. So maybe on one side, maybe the next thing here is just the edge of the paper so there's no crease there. If this was it, then I would say this thing is not mingling because this distance is bigger than this one and this distance is bigger than this one. So these m's are really clustered together. They're not mingling with the edges, so that's bad. And in fact, this thing is not flat foldable for the same reason that one is not.

But if, for example, I put a valley crease right here, then this is closer than that. I mean this distance is smaller than that one. So I call that mingling because at least on the right side, the m's are mingling with the v's. On the left side, they're not mingling, but right side's better than nothing. I need this to hold on at least one side. Could hold on both.

I claim this thing is flat foldable. Is that true? Yeah. Maybe. It doesn't look so true. I think I probably need some condition on the v also. Well, we'll see what happens in

the proof. That's more exciting. Usually the proof is easier than the example. Oh, I see why. There's a problem here. I think I need to say "forever." I'll say what I mean by that later. The proof will still work, but it's not such a pretty property.

The proof is in three parts, basically. First part is if you're flat foldable, then you are mingling. Certainly, if I want these all to be equivalent, it should be true that the first one implies the third one. This is actually really easy to prove.

Suppose it were not mingling. Suppose I had some crease pattern that violates that constraint. What that means is I have a bunch of m's here, I don't know how they're spread out. Then I know that there's a big gap until the next crease, something like that. Well, let's think about-- there aren't really that many ways to fold a whole bunch of m's in a row or a whole bunch of v's that's symmetric.

You pretty much have to spiral around. You're always turning the same direction. If it's flat foldable, which is what we're assuming here, it's got to look kind of like that. In fact, it has two ends, so it could actually make two spirals. But if you think about what happens there, this last part at the end of the spirals, that's this segment. That's when you possibly turn around the other direction. Maybe this is the valley.

But take a look at what's going on here. This segment is tucked inside the spiral. If it's super long like it is here, if it's longer than the previous portion-- let me label this. This is x, this is x, this is y, this is y. That looks really bad. x has to be smaller than y. Otherwise, it's going to penetrate right through that corner.

In this case with the double spiral, double spiral all the way, what does it mean? I guess no one knows double rainbow. Bad stuff. In the double spiral case, in fact, this end must be shorter than this one and this one must be shorter than this one. So in this case, you must be mingling on both sides, but it doesn't have to be that way. If it's just a single spiral, this last segment can be really long. That's OK.

It's much longer than this one, but the inner part of the spiral must always be smaller than the next segment of the spiral, and that's exactly what mingling says. Could be on the left, could be on the right, you don't know. But on at least one of the

sides, you must be mingling. And therefore, if you're going to have a flat folding at all, you must be mingling. Clear? Question?

**AUDIENCE:** When you say a maximal sequence, what do you mean?

**PROFESSOR:** Maximal means that you can't make it any bigger locally. I can't add any more creases and still have a contiguous bunch of mountains. It's a good question. In mathematics, there's maximal and maximum. Maximal means locally biggest, maximum means globally biggest.

So I don't mean take the biggest sequence of contiguous m's you can find. I just mean take some contiguous sequence of m's that you can't make any bigger on either side. I just mean that it's surrounded by valleys. I want this to hold for every contiguous sequence of m's.

That's one of those funny math technical terms. Other questions? So that's the first thing. Somehow I want to connect all these properties together.

The next part is that if you're mingling, either there's an end fold or there's a crimp. At least one of those two things must be possible. So this would be good news because if I'm flat foldable, then I'm mingling. If I'm mingling, then there's a fold I can make. That seems like progress.

Then I have to show that I don't mess anything up when I make that fold. But if I could keep going in a circle like this, then I would keep being able to make folds, and then eventually I would fold the whole thing because there's only end folds at the end. I'll have made them all. It's not that much trickier, I guess. It involves parentheses. That's one way to write it. Yeah, question?

**AUDIENCE:** So right now, the main property is finding a way that [INAUDIBLE] two or more?

**PROFESSOR:** That's right.

**AUDIENCE:** And there is an example that [INAUDIBLE]?

**PROFESSOR:** Yeah. Just ignore this. It'll be easier to think of it that way. I'm going to use mingling

to prove this property, but this is not equivalent to those properties. You need to add "forever," but that's too confusing at the moment. This is a definition of mingling. We're going to use it as a separate thing, and I'm going to prove these two things are equivalent. Then I'll be correct. Correct is always a good thing.

So what I want to do is for each of these mingling sequences, for each maximal sequence of m's or v's, I want to write down two symbols. I'm going to write a curved left parenthesis if it is left mingling, meaning I have all these mountains, let's say. Draw it in a reasonable way. Then I'm happy on the left side. I'm mingling with a v with an m fold on the left side. I know it's either left mingling or it's right mingling. I'm assuming mingling here.

I will write a square bracket if it's not. Then I will write a curved right bracket if you're right mingling, which is just the same picture but on the other side. And I'm going to write a closed square bracket if it's not.

Then I claim that parentheses sequence has some nice properties. So for example, this is a string that I get. It might look like this, maybe that. Who knows? It looks like something. I know that every pair, at least one of the two sides is curved. I can't have them both be square because then you're not mingling at all. I know I'm always mingling left or mingling right, so it's something.

Now, one thing that's really good is I have no double square bracket. That's impossible. I also know that if I had something like this, what does that mean? It means I have a bunch of mountains over here, then I have a very nearby valley, and then a bunch of valleys over here, and I know that the mountains are right mingling, which means I know that  $x$  is bigger than  $y$ . That's what this symbol means. It means to the right of the m's, I've got a small gap, and it means to the left of the v's, I also have a small gap. That means  $y$  is also bigger than this  $z$  distance.

**AUDIENCE:** Smaller.

**PROFESSOR:** Thank you. Smaller. I'm too tempted to combine them together, but it's a funny looking thing.  $y$  is smaller than both  $x$  and  $z$ . So then what do I do? Crimp.



That was sort of the definition of a crimp, that at least if I took this region  $x$  and copied it over here, those were empty of creases. And now we're saying that there's a big gap here. I guess technically it's bigger than or equal to if you said all these things right. I said "nearer," but I meant non-strictly nearer. It's enough to do a crimp.

So if I find this, I'm happy. The other thing that I'm really happy with is if I have an open paren at the beginning, a curved parenthesis, open parenthesis, at the beginning of the string, or a right parenthesis at the end of the string, because that means I can do an end fold. That means that you have a very short segment at the very beginning so you can just fold it over. This means you have a short segment at the end. Just fold it over.

So as long as I can find one of these three things, I am golden. I have what I want, which was an end fold or a crimp. How do I prove that I have one of those things? The easy way is by contradiction. Just imagine if you didn't have any one of these things, it's pretty forced what you have to have in your string because you know you must start with an open square bracket. You also know that you cannot then immediately have a closed square bracket, so you must have a closed paren.

Now I know that I cannot have an open paren because then I'd have this, and that would be good for me. So if I want to be bad, I have to have an open square bracket, but then I'd have to have a closed paren, and then I'd have to have an open square bracket, and so on. In the end, you will have a closed paren at the end, and that means you can make an end fold.

So either something failed in here. Maybe I had two parens together-- then I would have gotten a crimp-- or I can keep going all the way through here and get a closed paren at the end. In any case, I get at least one of these three things happening, and so I get either an end fold or a crimp assuming I am mingling. And instead of assuming I am mingling, I can just assume that I'm flat foldable, and then I get from this chain of implications that there is an end fold or a crimp.

OK so far? So that means I can make at least one fold. If I am flat foldable, I can

make at least one crimp or end fold. Now the question is, can I keep going? For that, we need one more property.

If I make a crimp or an end fold, I want to prove that that preserves flat foldability. What that means is if I start with something that's flat foldable and then I make a crimp or an end fold, I will still be flat foldable. This is the property I need because it will let me close the chain. I'm flat foldable, therefore I'm mingling, therefore I have an end fold or a crimp, so I make it.

This property says that I will still be flat foldable, therefore number one applies again. I'm still mingling, I can still find a fold, and I'm still flat foldable, so I'm still mingling. I can still find a fold. I can keep going. I will never get stuck because I'm guaranteed if I started flat foldable, I will be flat foldable forever. I will also be mingling forever.

I realize now but apparently have not realized before, it's not enough to be mingling just at the beginning. You have to be mingling through every step there. In fact, what we're showing is that you're fat foldable at every step, which implies mingling, but mingling is not quite enough. You'd have to check it all the way through. That's why I stated the theorem wrong.

How do we prove this theorem, this claim? I'm going to do it proof by picture. It's not always OK, but it will be OK. Trust me. We are assuming that we're flat foldable and we're assuming there's a crimp or an end fold. Actually, we know that that exists. I'm going to think about the crimp case because end fold is even simpler.

See, here's the thing. I would really like to fold some crimp like this, let's say. This is an m, this is a v. What I'm claiming is I'm allowed to make that crimp first and then do everything else, but conceivably, there's some flat folding that doesn't make this crimp first. I mean, it must fold this mountain and it must fold this valley, but at some point. We don't really know when. Somehow it all simultaneously collapses into some flat folding.

Now, this thing should be flat. It should line in a line. I'm stretching it out so you can

see it. It's a segment that we're folding, so maybe it does this, some crazy thing. Maybe this part does something crazy here, goes there, whatever. We know it's not crossing and it's sort of horizontal, but it could be an ugly mess. And in particular, in this picture, it does not look like this crimp was done first-- maybe embolden the crimp here-- because there's this stuff in the middle.

Now, we know that there's no creases in here. We know the earliest we can turn is at the ends of the crimp. But there could be stuff that comes in the middle here. What do I do? I just move the stuff out of the way. Let me do this the right way. Got to do this a little bit carefully, and unfortunately, my picture is so messy, hopefully you can see it.

So I'm going to take this part, which is inside the crimp, I'm going to move it down here, just lower those layers down to here. Now I know locally, there's a fold right here. And so if I take all the stuff that's in here and move it down here, it will still be non-crossing. And then I take this stuff and I move it up here.

Equivalently, you can think of me crushing this down and crushing this up, but it's actually important the direction these things go. I could not put this stuff up here because I don't really know how long this goes. It's not valid for me to try to put it up. I could collide with something. But I know that there's a crease right here, so I can put it right over here. I can take this stuff and put it right over here. This is a weird kind of manipulation of folded states which we don't normally do, but here we need to do it.

After I've done that-- is there any hope of me redrawing the picture? Maybe. After we've done that, we'll have the crimp. It will be nice and tight. And then we'll have this finger here, and then it goes up here, up this anvil. Then this part, instead of going inside, actually goes in here, up there, something like that, and that's the end. Then this is the other side. Other side wasn't too complicated, I think.

But you can check. I guess now it's down here. But you can really just do the shifting locally and it doesn't mess anything up. I'm going to wave my hands because it's a bit messy to really prove that. But if you do that, boom. Now this crimp looks like it

was actually done first. You can see that this folded state could have been made by first doing that crimp and then folding the rest. You can treat this as if the paper was fused together here, and that will be a folded state of that smaller piece of paper.

So even though we were shifting creases closer to each other and it looked a little scary what's happening, we know from this construction if you could fold the original thing, you can fold the thing even when the crimp got fused together. Therefore, crimps are always safe, and by an even easier argument, end folds are always safe. I already gave that argument, basically. So a little crazy, but that proves that you preserve flat foldability by going in this chain  $n$  or  $n$  over 2 times, you will fold everything if you were originally flat foldable. If you're not, you will get stuck because otherwise, you would find a folding, and so on. Clear?

I'll just mention you can generalize this to folding two dimensional maps with simple folds. I should say, of course, an end fold is a simple fold. A crimp is two simple folds. You can first fold along the mountain, you won't get any collision, and then fold back on the valley. That's a simple fold.

So this is saying, in fact, something is flat foldable in one dimension if and only if it is flat foldable by a sequence of simple folds. As long as you stick to crimps and end folds, it doesn't matter what order you do them in. That's not true as soon as you go to two dimensions.

So for example, just for fun, do I really want to draw the whole thing? I guess I've committed. There's a couple examples in the notes that you can take a look at, and I encourage you to make one. This can be folded flat but not by simple folds, and you can see that it can't be followed by simple folds because there's no fold that goes all the way through. This line is valley, mountain, mountain. There's no line that's all mountains or all valleys.

But you can use this structure to understand at least which rectangular maps with horizontal and vertical creases can be folded flat by simple folds, and the answer would be no here. All you do is you say, well, I know if there's any fold, I must have a mountain or a valley that goes all the way through.

So imagine here's your rectangle. There's got to be something that's valley all the way. Now, if you think about a crease that goes the other direction, these are two valleys. You can show, just by playing around with a little pattern like this, you cannot have these two both be valleys because that would be all four valleys is bad. If you try that, doesn't work. Or it also can't be both mountains. You can't valley, mountain, valley, mountain in order. That doesn't fold.

So in fact, these have to be different. One's mountain, one's valley. I don't care which is which, but I know that this line actually cannot be folded by a simple fold. It's got a valley and a mountain.

So in fact, the picture has to be I have a line like this. Maybe I have a few lines that are mountains all the way or valleys all the way. And any horizontal line I can't fold. I've got to finish folding these guys first. That means I have a one dimensional problem. So I apply this algorithm to see whether that one dimensional thing folds.

Once I've folded all those vertical things, solved that one dimensional problem, I will then have a horizontal problem, and then I'll have a vertical problem. You just keep doing this, and you can show in the same way you always preserve flat foldability. You just keep doing crimps and end folds, and this will fold your map if it's at all possible by simple folds, which most ones that are in production are foldable by simple folds. Question?

**AUDIENCE:** Yeah. If we go back to the example where [INAUDIBLE]?

**PROFESSOR:** Yeah.

**AUDIENCE:** [INAUDIBLE] mountain, it would be reversed and then it would be foldable [INAUDIBLE].

**PROFESSOR:** Yeah. This is good. This is the case that could work. This is foldable if you try it out. You first told here, and then the valley falls onto the mountain but it turns upside down as it folds, and so they actually nest nicely inside each other. It's confusing because it's inverting. This is the one valley case, but my point is you have to fold

this before you fold that. There's a little more details in the notes here. You can make this into a really fast linear time algorithm if you use fancy data structures, but that's beyond this scope, and that is the end of lecture two.