MIT BLOSSOMS INITIATIVE

"Taking Walks, Delivering Mail: An Introduction to Graph Theory"

Karima R. Nigmatulina
MIT

Section 1

Hello and welcome everyone! My name is Karima Nigmatulina, and I am a doctoral student at the Massachusetts Institute of Technology in Cambridge, Massachusetts. I'm excited to tell you today about a branch of mathematics called graph theory. You probably haven't heard about it before because it's not really algebra and it's not really geometry and it's not really arithmetic. It's something completely different. But, it has many, many applications and is very useful so I hope you enjoy the lesson.

Let's start back in the beginning when graph theory was born. We have to take ourselves back to the 1700s, to the city of Königsberg. Königsberg was a wealthy trading city in Prussia, and you can see on your screen right now a map of the city of Königsberg from that time. Königsberg was divided into four parts by a river called Pregel. The top part of the city was called Alstadt. The middle island part of the city was called Kneiphof. The lower part of the city was called Vorstadt, and the right middle part of the city was called Lomse. These four city parts were connected by seven bridges, and these bridges are shown in red on the diagram that you can see here right now. So here is what the city of Königsberg looked like in the 1700s. So the citizens of the city really enjoyed their city and they would walk around the city a lot. They created a game for themselves. The game was: could they start in one part of the city, walk around in such a way that they crossed all the bridges exactly once - not more, not less - and come back to that original spot where they started. The citizens of Königsberg weren't able to come up with such a route. But they didn't know if they just couldn't do it or if such a route didn't exist. And this is the problem that we're going to start with today. Do you think that such a route exists? Can we start in one point of the city, walk around, cross all the bridges exactly once and come back to that original starting point? Take some time with your teacher. See if you can come up with such a route and please let me know if you think that you can, or if you think that you can't. Tell me in five minutes. See you soon.

Section 2

Welcome back everyone. Were you able to come up with such a route around Königsberg? Well, luckily for the citizens of Königsberg they had a very brilliant mathematician who helped them out and solved this problem. His name was Leonard Euler.

Now, probably the first approach that you took to solving this problem was you decided to go through and figure out all the different routes possible through our city. You probably started in different places and then just routed through. That is an approach that would work if you could really go through and enumerate every single route. But think about what would happen if the number of bridges was much more than just seven. What if we had 20? What if the number of land masses that we had was more than four? In that case this problem becomes really, really tedious. Euler was actually much more ingenious than that. He came up with a completely different approach to that problem than the enumeration one.

This is what he proposed. He proposed that every single land mass be represented as a node. And the node would have a capital letter attached to it to represent that node. Now in addition to land masses, we have bridges right? And the bridges would be represented by arcs. And those arcs would have a lower case letter to represent them. So we have capital letters, nodes, which are land masses, and we have bridges, arcs, lower case letters. Now how would we represent Königsberg in this case? Well, we would represent Königsberg as such. We have four nodes with capital letters, A for Altstadt, K for Kneiphof, V for Vorstadt, and L for Lomse. And then have seven arcs connecting these land masses, connecting these nodes, and those represent the seven bridges in the town of Königsberg.

Now one thing that we can do, is we can represent this as a graph. So this is a graph. Now graph theory says that we have nodes as we said, and we have arcs. What are the different nodes that we have in Koningsberg? Take some time to think about it. Talk to your neighbor. What have you come up with? You probably came up with nodes A, K, V and L. Now in addition to nodes, we have arcs. What about the arcs? What are the different arcs in the city of Koningsberg? Talk to your neighbor again, come up with the different arcs. Hopefully you were able to come up with arcs a, b, c, d, e, f and g. Now together the nodes and the arcs represent a graph. And oftentimes the notation that we would use for a graph is along the lines of G, which stands for a graph, is composed of nodes and arcs. So this is your first taste of graph theory.

Now the observation that Euler made is really an ingenious one. He used this graph and said, “Well first of all let’s look at all the different nodes. All the different nodes have a different number of bridges or arcs attached to them. We can call this number of bridges a degree.” So the degree, for example, of node L is three. We have three bridges coming out of or attached to L. What about for node K? What’s the degree of node K? You’ve probably figured this out on your own, but just to make sure, the bridges that are connected to node K are d, e, f, b and a. That’s five different bridges, so the degree of node K is five. The really brilliant observation that Euler made is: if every single node within the graph has exactly an even number of bridges or an even degree, then and only then will such a route around the city exist.

So now that you have that observation, now that you know that you need an even degree for every single node, you should be able to figure out whether Königsberg does or does not have such a route around it. By the way such a route, in honor of Euler is called an Eulerian tour or sometimes an Eulerian circle. That’s the first thing that I want you to look at with your teacher. Does an Eulerian tour exist around the city of Königsberg? And number two, I have a bit of a challenge for you. What do you think would happen if we could start and end our tour in two different places? Does the city of Königsberg have such a path that would start in one place, move around, cross all of the bridges exactly once and end up in another place? Think about it, and I’ll see you in a few minutes.

Section 3

Were you able to come up with a tour around Königsberg crossing all the bridges exactly once? Well, you probably weren’t able to because as you probably just figured out that all of the nodes have an odd degree, and because of that such a tour does not exist.

Now the other question I asked you is: what if you could start and finish in two different places? Would such a path exist in that case? Well, here’s the observation that Euler made about such a path. He said that you could start and finish in two different places and cross all of the arcs exactly once, if exactly two of the nodes have an odd degree and all the other ones have an even degree. If you remember our Königsberg graph, we had four nodes and all of them had an odd degree. So in the case of Königsberg, neither a path that would start and end in two different places, nor a tour that would start and finish in the same place, exist.

Now what about the logic of this? Does this make sense to you? Well let's think about this. Let's assume that we can represent one of the land masses by this piece of paper, so we have this land mass. Now within this land mass we have several bridges attached to it. I'm going to blindly pick up a piece of paper. There are pieces of paper one through ten in this hat. I'm going to pick out one and that's going to be the number of bridges that we have attached to the land mass. The piece of paper that I have picked out is five, so we have five bridges attached to our land mass. Let's draw them. So we have one, two, three, four, five. Let's go through this slowly. What if you come in on this arc? Well, if you come in on this arc and cross that bridge, then you could leave on this arc and then you'd cross it. Eventually you'd have to come in on this arc or any of the three left over and then you leave on this one. Well, then eventually you'd have to come in on the last arc. You come in and then you get stuck. You can't leave the node. Well, the only way that you could leave the node is if you actually had another extra arc coming out, another bridge, so you could leave that node. Now we actually have six bridges, so that's an even degree. That is the logic behind Euler's claim. Now think about this with your teacher a little bit. Try to figure out why it is that for the case where you start and finish in two different places you are allowed to have two nodes that have an odd degree. Does it make sense to you? I hope it does. But after a few minutes with your teacher I'm sure you'll have it all figured out. See you soon!

Section 4

Welcome back. You were probably able to figure out the logic of the Eulerian path where you can start and finish in two different places. The reason you could have two nodes with odd degree is because you could start at one of the odd degree nodes and finish at the other odd degree node, so that solves the problem. But now that we can move on from that, let's generalize the problem and see if we can come up with an algorithm for coming up with such an Eulerian tour where you actually start and finish in the same place for any given graph.

What we're going to do is we're going to look at the following graph that we have right here and actually go through it and use the examples to come up with the algorithm. We're going to start with node C and finish up at node C, so that's going to be the algorithm. But before we proceed, here is a quick question. Does this graph have an Eulerian tour? Well, we have five nodes and every single one of the nodes has four arcs attached to it, four arcs coming out of it. Now those four arcs mean that each single node has a degree of four. Four is an even number so we're all set. And let's get started.

We're going to start in node C. By the way, you could start in any single one of the nodes. I just picked node C because why not? OK. We're starting at node C. In this case we could start at node C and go to B, A, E or D. Any one of those would really work. I'm just going to pick C to A, so we're going to cross the following arc. And as you see I actually crossed it off to remind myself that we already went from C to A and that we don't need to cross that arc over again. OK so we cross it off. And the other thing that we're going to do is we're going to write down that we crossed arc C-A. That was the first arc that we crossed, the first thing that we did. So here we have it written down. Now that we are at A, we could go to D, to E, or to B. Again, up to you. We could pick any one. Why not just go to D, why not? OK, so we crossed off arc A-D. And write that we go from A-D on our note keeping list there. Now that we're at D, we could go to B or C or E yet again. Just pick any one. I pick B. So now we're at B and we write that we crossed D-B next. You probably are going, "I sort of have the hang of this." Well let's go from B, then we can go to C, yet again just keeping track of everything that we're doing. And then from C we can go to D. So we crossed C to D. And then from D as you see we actually don't have any choice. We have to go from D to E. Well, no choice, that's fine. We're still doing well, so we cross from D to E and write it down. Now that we are E. We have several choices. We can go E to C, E to D or E to A. Now here's a very important thing that you need to notice because this is really the main trick of the algorithm. If you cross from E to C, look where you'll end up. You'll end up in C and get stuck because all of the arcs at that point attached to C will have been crossed off. That means that you can't cross E to C. This type of arc that gets you into a stuck position is called an isthmus. That's the lesson of the algorithm. Never cross an isthmus before you are really complete with the algorithm. So, no crossing isthmuses. Well, that's all right. We're not going to cross that isthmus, we're not going to go from E to C, but we're just going to go from E to A. Why not? Let's go E to A. Write it down. From A to B no choice, so that's the one that we'll cross. Sounds good. From B we go to E, yet again because we have to. From B we cross to E and then lastly we're going to cross from E to C and we're all set. As you might remember E to C was our isthmus arc before. The lesson here is that you do cross isthmus arcs. You just cross them once they stop getting you into a stuck position when you are ready to be done with the algorithm.

OK. So we've crossed all of our arcs and we have this notation written down. The main thing that you want to do right now is you want to be able to write down what the actual path was. So let's start from the top. We went from C to A, then A to D, so we have C-A, A to D, D to B, B to C then to D, then to E, then to A, then B, E, and C. So that is our tour. And notice we started at C and we ended at C. And that's pretty much it.

I have a little bit of a problem for you to do on your own now that you feel really comfortable with this algorithm. Take a look at the screen right now and you should be able to see a graph that I want you to try and figure out. First of all does an Eulerian tour exist for this graph? If it does, actually come up with that Eulerian tour. How about you try for node A and then maybe try for another node. You can pick one that you like and go for it! Hopefully you'll figure it out and you'll feel very comfortable with it and then we can see you soon!

Section 5

I hope you enjoyed learning about the seven bridges of Königsberg. You actually probably are curious about what happened to Königsberg and its seven bridges. Unfortunately in World War II, the city was bombed and many of the bridges were destroyed. Along with the different layout there's actually a different name for the following city. It is now called Kaliningrad and it is part of Russia, that's actually where I am from. The city actually still lives on. It's a great, wonderful city, and you can always look up and find out more about it in maybe some of your history or geography lessons. But the main lesson for today was that this was a city that was the birthplace of graph theory.

Now you have two choices. You have an option here right now. Either I'll be seeing you in a couple of minutes after you stretch and kind of move around a little bit, or I'll be seeing you tomorrow and we'll move on to a topic that's very similar and build on the lesson plan that we covered today on Eulerian tours and take this a little bit further into graph theory. So either I'll see you in five minutes, or I'll see you tomorrow!

Section 6

Welcome back. Whether it's the same day or whether it's the next day, I hope you enjoyed learning about the Königsberg bridges in our previous lesson. What we are going to do now is we're going to take that knowledge and we're going to apply it to a slightly different type of problem. This new type of problem is still part of graph theory, but it takes and uses some of the information that we learned from the Königsberg bridges and applies it further. Hopefully you enjoy it.

OK, we're leaving Königsberg and we're fast forwarding several hundred years. Take yourselves to China in 1962. There was a Chinese mathematician in 1962. His name was Mei Ko Kuan., and he came up with the following problem that postmen face all over the world. For a postman there is a post office and there's a route that he needs to cover. For example, we have a village and there are all kinds of roads within this village, and the postman will have to cover every single one of the roads to deliver the mail. What he'll have to do is he'll have to start at the post office and he'll have to finish at the post office after he's done his route. The question that Mei Ko Kuan came up with was, "What is the shortest route around the village covering all of the roads and starting and finishing at the post office?" By the way, if you're wondering why is this such an unusual problem and is there anything particular about it that it's a Chinese Postman problem? No, not really. All postmen all over the world pretty much face the same problem, but since a Chinese mathematician came up with it, this problem was called the Chinese Postman problem.

Now let's apply this problem to a specific example. Let's assume that we have the following village. There are nodes and arcs just like in other graphs that you saw in the Königsberg problem. Here we have the nodes and arcs representing slightly different things, so pay attention! OK. We have one node, in this case we'll call it node A, that represents the post office. That's the location where the postman will have to start and that's where the postman will have to finish up. Now we also have a whole bunch of other different nodes. Those nodes represent the intersections in the roads where postmen can choose to go one direction or the next. In addition to these intersections and the post office, we have arcs like before, and those arcs represent our roads. Here are all of the different roads that the postman will have to cover while traveling around the village and delivering his mail. And the new addition from the previous problem is that we have numbers associated with arcs. These numbers describe the length of the road going from one intersection to the next or from the post office to the intersection. So for example, going from D to intersection F takes a length of four. OK. So that's the problem formulation. Yet again, let me review the question for you. What is the shortest route that we can take around our village starting and finishing at A and covering every single arc at least once?

You may think that this is sort of a little bit similar to the bridges of Königsberg problem and you're right. If there is actually an Eulerian tour for our graph, then we have been able to come up with a solution that's just wonderful and fantastic because you won't be crossing any arcs twice. But what happens if an Eulerian tour does not exist? Well, in that case you will be crossing some of the arcs twice. Not a problem, but what we want to do is we want to figure out which arcs are the ones we can cross twice without making our route too long. OK, so here's the question that I have for you. Can you come up with a solution to this Chinese postman problem for this graph? By the way if you come up with a minimal bound for the Chinese postman problem, you come up with an obvious solution of how long this tour would be, the minimal one, the bound for it? After you do that, try to come up with a solution for this problem. If you aren't able to come up with it, that's all right because we'll be doing that in the next section.

Section 7

Welcome back. Were you able to come up with the shortest path for the graph that I gave you? Well if you weren't, don't worry. We'll be doing it right now. I'm going to introduce an algorithm to you that will work for any type of Chinese postman problem. But, before we can do that we'll have to come up with a few insights.

First of all, the main thing that you're going to do is in the previous graph that I gave you, you might have noticed that there is no Eulerian tour. So it's a kind of a difficult thing because you'd have to figure out which arcs you're going to cross twice. Well, the algorithm that we're going to come up will tell you which arcs you have to cross twice. And what we'll be doing is we'll be altering our current graph, to a new type of graph by adding a few arcs or figuring out which arcs we're going to cross twice. That way we can sort of figure out what the Eulerian tour would be for the new graph. The idea is you have a current graph. You'll be altering it by adding artificial arcs to come up with a graph that has an Eulerian tour. What that will require doing is adding artificial arcs that will first of all have to maintain the evenness of the nodes that are already even. And number two, they'll have to change the nodes that are currently odd to even type nodes. That's the overview of what we're going to be trying to do. But, there are a few intuitions and a few insights that you need to kind of get along the way, to make sure that you understand what's going on within this problem.

First intuition, and this may come out sort of like something out of the blue, but the first insight that you need to see is that the number of odd degree nodes is always even. That's a bit of a mouthful, so let's actually write it down and see what it is that we're trying to say. What we're trying to say is the number of odd degree nodes, so I'll put a shortened version of degree. So, the number of odd degree nodes is even. This is what we're going to be trying to figure out why. But that isn't the main conclusion that we're going to be going about.

OK. There are a few sort of general insights that you need to see along the way to make sure that this really all comes together. First of all, look at the degree of overall graph, and add up the degrees of all the nodes in the overall graph. First thing, that number will always be even. The first conclusion that I'm going to try and prove, the first step, is the number or the sum of all node degrees is even. Does that make sense to you? Why do you think I say that? Well, here is a way to understand it. Think of an arc, one arc that's added to a graph, so let's actually draw an arc. Well that arc will have a beginning and an end or a beginning and an end, whichever direction you want to go in. But it will be attaching to two nodes. The degree that it will be giving to the whole graph is that it will be adding one degree here and one degree there. So the total number of degrees added will be two. Every single arc adds an even number - two degrees to the overall graph. That's why it makes sense because regardless of how many arcs we have, the degree of the total graph will always just be the number of arcs times two, an even number. Hopefully I have you convinced about the first part of this.

Now the next conclusion is that the sum of odd degree nodes is even. So if you add up all the odd degree nodes' degrees, what you'll get is still an even number. Let me repeat that statement one more time. For odd nodes, after you add up all those odd nodes' degrees, you'll come up with an even number. Why is that? Well, the insights that you need to use are a combination of two things. The first one is that the whole graph has an even degree total. And the second intuition is that whatever the even nodes contribute is even, the total degree of the even nodes' will still be even. So you have a total graph degree even and you'll be subtracting all the nodes that are even. An even number minus an even number gives you another even number. And that even number result is exactly the sum of the odd degree nodes' degrees. Yes, quite a mouthful. But hopefully it makes sense to you when you think through it.

Now what we're going to do next is we're going to use this intuition and come up with an algorithm that works to solve the Chinese postman problem. Hold onto your seats! We're going to move on forward.

Now that we have the intuition about the number of odd degree nodes, what we're going to do is we're going to use that knowledge to come up with the algorithm. What I'm going to do is I'm going to give a sneak peek of what the algorithm looks like. OK, here it is. We have four steps to our algorithm and what we're going to do is we're going to apply these four steps to our graph right here. Let's read step number one. Step number one says, "Identify odd degree nodes." Pretty easy. Pretty straightforward. Looking at our graph, which are the odd degree nodes? This should be a real piece of cake for you right now. Let's see is A an odd degree node? It has three arcs, so yes it is. And let's mark it as such, let's draw a little star next to it. What about other nodes? E? E is an odd degree node as well so let's draw a star next to that one as well. C? No, that's an even degree node. It has four arcs, so we're all set there. What about D? That also has four arcs attached to it, so yet again even degree, all set. No problem. Now B has three arcs. Three arcs, odd degree, let's draw that star next to it. And lastly, what about F? Well F has three arcs and so it's an odd degree node and yes we're going to draw another little star next to it. By the way, here is a quick thing that you should notice. Look at the number of odd degree nodes that we have. It's even. We have four odd degree nodes. Four is an even number and that's something that we concluded just before. All right. We've done step one.

Moving on to step two. Let's read step two. Step two says, "Find a minimum length pair-wise match for the odd degree nodes." So we have four odd degree nodes. A, E, D and F. By the way, notice that if we had an odd number of these odd nodes, we'd be in trouble because then we couldn't do a matching; we couldn't pair them up. One would be stuck. This is where we use the intuition and the knowledge that we concluded before. All right. We're going to do a matching of our odd degree nodes. Well, how do we know which one is the best one? Let's just try all of them and see which one is the minimal matching.

OK. Let's try one matching: A-B, E-F. Let's write it out. So if we match up A-B and we match up E-F, what we're asked to do is we're trying to figure out the minimum length matching. Well what is the length of a matching? I actually haven't told you that yet. So listen up, here's what it means. Between A and B, what is the minimal distance route you could take to get from A to B? Well, you could go directly A to B on that arc of length 8. You could also go from A to C to D to get to B. That route has a length of 8 as well. If you look at all the other possible routes that you could take to get from A to B, they will all have a length of greater than 8. So 8 is our minimal length to get from A to B. So let's write that down. What about the minimal length from E to F? Can you see it? Do you think it's E to F directly a length of 9? Well if you say that no, that's not the length, then you're right. Because actually the shortest path from E to F is going through E, C, to D to F. Going 2+1+4 and that gives you a length of 7. And 7 is shorter than 9. So the minimal length for the E-F path is actually 7. Now the length of the total matching is going to be the sum of 8 +7 and that's 15. So we have one that's a length of 15. That's one matching.

But we have more matchings, right? We could match up A&E and B&F. What about that one? OK. Well, A and E and B and F. What's the length from A to E? What's the shortest path there? Can you see it? Hopefully what you see is that the shortest path from A to E is A to C to E and that's a length of 5. So 3 +2. So let's write it down. That's a length of 5. What about B to F? What's the shortest path there? OK. Well here it's actually pretty straightforward. You just go from B to F directly and that's a length of 5 as well. And 5 is much better than going through D or taking any other longer route. OK. So we have 5 + 5 that's the matching A-E, B-F has a total length of 5 + 5 = 10. So we have another matching with a length of 10. Ten by the way is smaller than 15. So this was good. We found a matching that's better than the previous one.

Do we have any other potential matchings of the four nodes? Do you see maybe another one because there is one more. If you match up A to F and E to B. Let's write it out. A to F and E to B. Let's call this matching number two and this is going to be our third matching. What's the shortest distance from A to F? Well, we're going to take that diagonal and we're going to see that it's 3 + 1 + 4 = 8. And if you're not convinced try to come up with other routes and you'll see that that's the shortest route that you can take. So A to F has a length of 8. Now what about E to B? What's the shortest path there? Well if you again take the diagonal type of route, it's 2 + 1 + 4 = 7. And you can try and look at the graph and come up with any potentially shorter path but you won't be able to. So it has a length of 7. That's the shortest path from E to B. Now the length of that total matching? Well it's 8 + 7 =15. Now we have is three matchings of three different lengths. We have 15, 10 and another 15. What's the shortest one? 10. Ten is the minimum number of those three. That means that A-E, B-F is our best matching that we want to use. So let's underline it to remind ourselves that A-E, B-F is our shortest pair-wise length matching. We've done step two. Let's move on to step three.

Step three says, "For each pair of nodes add the arcs of the shortest path." So the pair of nodes that we are referring to is that pairing that we just did in step two as well as the shortest path that we used to figure out the length of that matching. Let's do it. The matching that we have that's the shortest one is A-E and B-F. What's the shortest path for A-E? As we said it has a length of 5 and it goes through C. What we're going to do is we're going to add an extra arc from A to C and another extra arc from C to E because that will represent that it was the shortest path. That's what step three tells us to do, so we've added the arcs there. Now we also need to add the arcs for the matching of B-F. The shortest path there is directly straight and what we can do is we'll just draw it in there. So that's the next arc that you are going to add. And that means that we've done step three.

Now here is an interesting observation that you should make. Look at all our nodes. In this altered graph all of our nodes are of even degree. Look. E had a degree of 4. A has a degree of 4. B has a degree of 4, F has a degree of 4. D did not change it's degree so it's still even, and C which was even before, had a degree of 2 added to it, so 4 +2 = 6 and that's its current degree and that is an even number. All of our nodes have an even degree. That means that there is an Eulerian tour in there. This is where you get to use the knowledge from the previous lesson of the Königsberg bridges. What you now need to do is step four.

Step four just says, "Find the Eulerian tour for our new graph." Within our new graph we know that there is an Eulerian tour because all the nodes are of even degree and we'll just need to make sure that we start at the post office, or node A and then after we tour around, come back to node A. The quick little thing that I want to tell you before we go on to a new type of question is the length of the Chinese postman problem at this point can be determined by adding up all the arcs from before, from our original graph, and you have all of these lengths marked in blue. By the way, if you actually just do that, the 6 + 2 + 3 plus all of the other numbers, you'll see that the length of those is 44. But we are repeating and going over certain arcs twice, or the ones that are the artificial ones. We know that the length of the artificial arcs that we added, these white ones here, has a length of 10. So what we have is 44 + 10 = 54. That means that the length of the Chinese postman problem solution to this graph is 54.

OK. Now we have come up with a solution to our example and hopefully you feel pretty comfortable with it. But, I'm guessing you want a little bit more practice. What we're going to do is we are going to solve the Chinese postman problem for the bridges of Königsberg. Remember the graph? It probably looks a bit familiar. The main changes that have occurred here are that now I've added the numbers on there and those numbers represent the lengths of all the bridges. OK. Take some time with your teacher and come up with a solution to the bridges of Königsberg using the Chinese postman problem for it. What is the minimal length that you would have to travel around Königsberg and cover all of the bridges at least once.

If you don't think that's enough practice for you, I have another example that you could do. You can refer to your slide that you see on the screen right now. That slide should describe the problem of a few more nodes. The post office here is located at node A, like before. Make sure that you can figure out what would be the solution to the Chinese postman problem for that graph as well. And hopefully that will give you a lot of practice and you'll feel really comfortable with this problem. I'll see you soon!

Section 8

Congratulations! You now known graph theory. It's actually often called network theory so that's another name for it. You may be wondering why do I really want to know about graph theory? Does it have applications anywhere outside the post office? It does. I promise. I told you in the beginning that it did.

Here are some examples. Some really basic similar examples to the post office problem are: you can have street cleaning or you can have snow cleaning or the garbage collection system. All of them are actually along the lines of the Chinese postman problem. You need to cover up all the different roads and you don't want to travel all that much distance. So you're trying to minimize distance while covering all the roads. Just like the Chinese postman problem.

But that's not the only problem in graph theory. There are a lot of others and there is many of these applications. For example, you GPS system, if you have heard about those, they route cars from one location to the next in the shortest way possible. They actually also use network theory or graph theory. Airport or airplane routing use graph theory as well. Even circuit chip creation uses graph theory. It's a really, really commonly applied branch of mathematics. So I hope you enjoyed learning about it. Well, at this point I wish you goodbye and good luck in all your studies! I hope you enjoyed the lesson. Bye!

Teacher Guide Section

Hello. First of all, let me thank you for agreeing to do this and I hope you enjoyed this lesson plan as much as I enjoyed creating it and hopefully your students enjoy it as well. What I'm going to try to do right now is try and explain what I had in mind for some of the pauses in between the different sections. Hopefully this will help you out in terms of leading the class. First of all, the material that we're going to cover within the whole lesson is called graph theory. It's will be a new branch of mathematics for probably most of the students. It doesn't really have that many prerequisites in the sense that it doesn't require any complex algebra or any complex geometry. More or less, it just requires a little bit arithmetic and a little bit of an understanding of what happens in mathematics, a little bit of mathematical intuition. So it's probably a good fit for anyone in high school.

Furthermore, what I'd like to add is that this lesson plan really has two parts to it. There are two problems that I go through. There is the Bridges of Königsberg problem and there's the Chinese postman problem. It's up to you and up to the students however you see fit; you could either have those done on two separate days and just do one section on day one and another section on day two. Or you could just decide to push right through the whole class. But in that case, it might be a little long. Maybe you can make the pauses a little bit shorter but it's really up to you and you can make that decision.

Within the first pause, after I introduce a sample problem, which is the seven bridges of Königsberg problem, usually the best thing to do with the students there is to try and give them some time with the graph, with a picture of the map and see if they can come up with a route around the city that will take them across every single bridge exactly once and finish and start in the exact same place. What I had in mind, maybe you could divide up the class into four different groups and each group could start in a different part of the city. One in Alstadt, one in Lomse, one in Vorstadt, one in Kneiphof and make sure that they have a sense that all these different locations could be a starting point. And after giving them a chance to really try and come up with a route, the other thing you could do is have the students take a vote on whether they think it's possible or whether it's impossible; whether such a route around the city does or does not exist. That was my general idea for pause one.

In terms of pause two, after I explain the ideas of what really allows for an Eulerian tour to exist, what the students can try to do is: first of all, see how it applies to the Königsberg city and that map there. In order to determine whether an Eulerian does or does not exist, you can just count up the degrees of every single one of the four nodes and you'll see that all of the nodes are an odd degree, and that will just lead the students to conclude that no, an Eulerian tour does not exist.

Afterwards there is a challenge problem that I ask of the students and the challenge problem is pretty much: if you can start and finish in two different locations, could you find such a tour within the city? This is a bit of a challenge. It will be a little bit of a reach for many of the students, but maybe someone will be able to come up with it and really see if they can have the intuition there. The main intuition that they could get, which would be really fantastic, is that if you start in one of the odd degree nodes and finish in another odd degree node, that you could have two odd degree nodes within your city. In that case you could have this Eulerian path which starts in one odd degree node, and then finishes in another and the conclusion there is you can have two nodes of odd degree. If the students don't really get there, I don't think it's a really big deal to push them through and make sure that they get it. I will be reiterating it within the next section, so hopefully more or less it will make more sense there.

Pause three, this is the pause that's really up to you. The students may or may not be confused about the concept that I introduced in that section, so just to make sure that they understand it. Why is it so important to have an odd degree node? Maybe you can almost repeat the exercise that I did within that section and hopefully that will clear up some of the confusion for the students. Maybe a simple example that you could do is: draw just a node with one arc coming into it, it will be very clear that you can either leave on the arc and never come back, or you can come in on the arc and then you can never leave. The goal is to drive across the point that if you have an odd degree node, you can either get stuck, or you can never get back to that node. That's the logic. Hopefully, this type of simple example can really clarify to the students the various concepts and they'll be very clear and happy with it. You could also do the same thing, just draw two arcs and make it really obvious that if you have an even degree it's easy to come in and to come out within those nodes.

OK, now moving on to pause four. I actually have it on the slide right here. What you will be asked to do is you will be asked to come up with the algorithm or trace through all of the arcs within the graph there. I have a sample solution for you, so you can go from A to D on the top bar. Then D-A, A-D, D-A. Within this specific example what I wanted to get across is that if you follow steps one through five, at step six, I wanted to point out that there is an isthmus. This is a bit of an unusual concept and maybe the students will feel comfortable with it by this point or maybe you can re-illustrate it at this point again. Here's the sample solution for you and hopefully you'll be comfortable with it. By the way this is not the only solution and if you decide to do another one, that's perfectly fine as long as you feel comfortable with it. So that's the pause for going through the examples.

At this point, within the next pause you could either decide to take that pause and wait until tomorrow for the next lesson, or you could push through and move on the Chinese postman problem. Again, it's really up to you.

If you decide to push through, regardless of whether it's the next day or not, within the next pause there will actually be another content based pause. Within that pause I'll be asking the students to go through and determine what is the minimal length path around the graph. First of all, the first question is: what is the minimal bound for the length of the Chinese postman problem for the example route that I gave them. What I wanted to convey to the students is the idea that you have to cross all arcs at least once. If you sum up the lengths of all the arcs, then that's the minimal bound that I was referring to. And the students can do that if they add up the length of all the arcs. I believe the length of all the arcs should add up to 44. Also it may be good to give the students a little bit of time to try and play with the graphs a little bit to see if they end up with a solution to the Chinese postman problem. Just let them do it. You can give them a hint that they can use a little bit of the intuition of the seven bridges of Königsberg problem and the Eulerian tour. But other than that, I'm not sure they'd be able to come up with an algorithm. But, it's really good practice to really challenge them and push them a little bit to see if they can come up with it on their own.

Now that will be pause five and then once we move on to pause six it will be the solution of two different problems. First of all, the problem that we're going to do, the simple one, if you think you're running short on time, is you could look at the problem of the seven bridges of Königsberg with the different arc lengths that you see on the slide. I have all the possible matchings listed on there because all of the nodes are odd degree. And the possible matchings are listed. The solution of it is presented on there. It's OK to say that there can be ties. As we see within this example there is a tie between the A-K-O-D matching and the A-V-K-O matching. And if you actually show it on the graph you'd be retracing the exact same arcs. Maybe that's something to point out, to make sure that the students don't get confused anywhere along the way. That's one of the examples that I've given them. If you have some extra time and you'd like to do another example to make it really clear to the students how to do this algorithm because it does get a little bit confusing, you can do the following example shown on this slide. Yet again, I pretty much wrote out the whole solution here. The potential matches of E-F, C-B, E-C, F-B and E-B, F-C. The main thing there is just to make sure that the students know how to find the shortest path between two nodes. And it's really important that they realize that sometimes it's not the direct connection, and sometimes you have to go through this indirect route and that can be the shortest path. That's the main lesson to take away for them there. And at the end you see the solution of the sample path of a sample Eulerian path is just the A-B-C-F-

E-G-E-D-E-A. Starting at A, finishing at A. It may be helpful to reiterate, yet again, that you start and finish at the exact same point.

And at that point it's really the completion of the lesson plan. Hopefully all of the questions of the students have been answered, and hopefully they enjoyed it. I definitely enjoyed teaching this material and I thank you very much for all of your help and all of your attention. Thank you very much! Bye.