M1Maths

G4-2 Networks

- graphs, vertices, edges, traversability
- weighted graphs, shortest paths

Summary Lead In Learn Solve Revise Answers

Summary

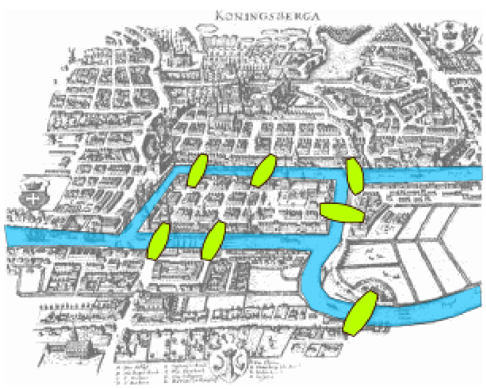
A network or graph consists of a number of points (vertices) marked as dots and a number of lines (edges) connecting the vertices. Graphs/networks have various applications.

The degree of a vertex is the number of edges connected to it. Odd and even vertices are vertices with odd and even degrees. A graph is traversable if it is possible to trace a path which goes along every edge just once each. To be traversable, a graph must have no more than two odd vertices.

Weighted graphs have a number assigned to each edge. These might represent distance, time, etc. On a weighted graph, the shortest path algorithm can be used to find the shortest distance, time etc. between any two vertices.

Lead-In

The Bridges of Konigsberg



In Northern Europe, between Lithuania and Poland, there is a small part of Russia separated from the rest of Russia. It is outlined in red on the map to the right.

The main town in this enclave is Kaliningrad. Back in 1736, being then part of German Prussia, it was called Konigsberg.

The river Pregel runs through Konigsberg and has a number of branches. In 1736 there were 7 bridges across the river as shown in the map to the right. A puzzle at the time was to find a way to cross all 7 bridges without crossing any of them twice (and without swimming or using a boat).

Images: Google Maps, Wikimedia Commons

Give it a go. See if you can find a route that crosses each bridge exactly once.

It took the German mathematician Leonhard Euler (pronounced "Oiler"), one of the most brilliant mathematicians of all time, to solve the problem. He solved it in 1736 and in so doing laid the foundation for a branch of mathematics called graph theory.

Graph theory doesn't deal with the graphs you would be familiar with (y vs x, *distance* vs *time,* bar charts etc. Instead it deals with a different type of graph which we will see below. The word 'graph' comes from the Latin word for 'draw'; both types of graphs are drawings.

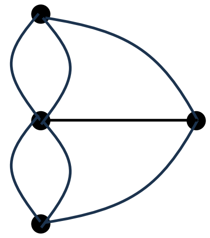
Another word for the type of graph used in graph theory is 'network', so in this context, 'graph' and 'network' mean the same thing. The two terms will be used interchangeably. This module gives an introduction to graphs/networks.

Learn

Networks/Graphs

In the map of Konigsberg, you can see that there are 4 bits of land separated by rivers. We can produce a simple model of the situation by representing these 4 bits of land as dots. Each bridge connects two bits so land and is drawn as a line connecting the corresponding dots.

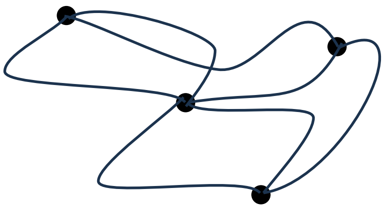
The result might look like this.



This diagram is a **graph** or **network**. Each dot is called a **vertex** and each line is called an **edge**. So the graph has 4 vertices and 7 edges.

The 4 vertices represent the 4 bits of land and the 7 edges represent the 7 bridges.

Note that the positioning of the vertices and the shape and length of the edges doesn't matter in a graph. All that matters is the number of vertices and the number of edges between each pair of vertices. So we could have drawn the network diagram like this:



Though of course it would be silly to do so because it makes it much harder to see how it relates to the map.

This distorted graph is **isomorphic** to the more sensible graph, meaning that they have the same vertices connected in the same way by edges. But that's probably not a word you need to remember.

Drawing the map as a graph doesn't really make the problem any easier to solve, but it does allow us to use a new technique which *will* make it easier to solve. We will now find this technique, but you will learn more and remember it better if you discover the technique yourself. So we will lead you to it through some questions.

To solve the bridge problem we need to be able to trace a route which passes over all the bridges just once each. On the graph this means drawing a route that passes along each edge just once without taking our pencil off the paper.

Copy the graph onto paper, grab a pencil and see if you can do that.

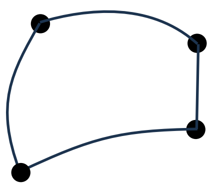
If you can do that (draw along every edge exactly once without lifting your pencil), then we say that the graph is **traversable**. Not all graphs are traversable.

The route we take through the graph is called a **path**. If the path goes along every edge just once each, then it is called an **Eulerian path**. ["Eulerian" is pronounced "oil air Ian" with the emphasis on the air.] So deciding if a graph is traversable is the same thing as deciding if it has an Eulerian path.

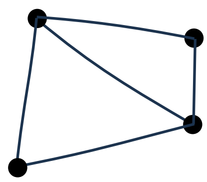
Practice

Q1. For each of the graphs below, decide whether it is traversable, i.e. if it has an Eulerian path.

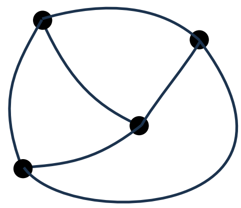
(a)



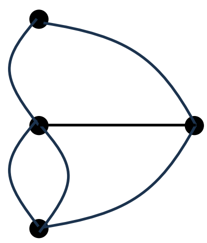
(b)



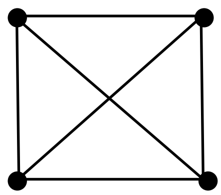
(c)



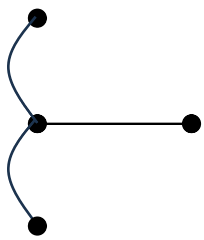
(d)



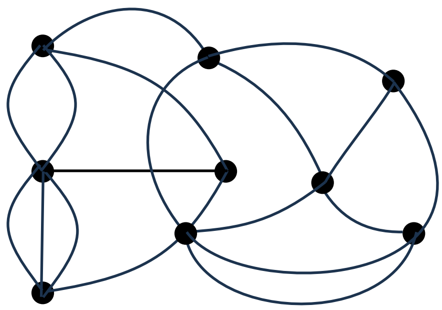
(e)



(f)



(g)



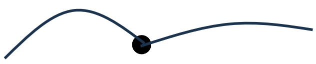
The last one of those might have been a bit of a challenge. But it is possible to tell that it is traversable just by looking at it and counting edges: there's no need to actually find a way to traverse it.

How can we do this?

If you would like a challenge that would help improve your mathematical prowess, see if you can find how this is done. If not, or when you have done it, read on.

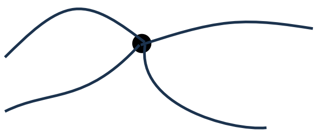
Deciding if a graph is traversable – the easy way

If a vertex has two edges coming from it like this,



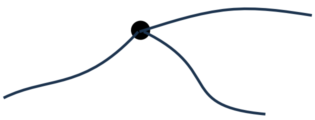
then the path can come into the vertex on one of the edges and leave on the other.

If a vertex has four edges coming from it like this,



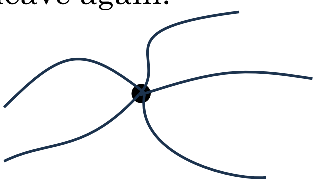
then the path can come in on the first edge, leave on the second, come in again on the third and leave again on the fourth.

But think about a vertex with three edges like this.



The path can come in on the first, leave on the second, come in again on the third . . . but then we are stuck. In trying to traverse the graph, we have to stop at that vertex. Or, of course, we can start there, then leave, return and leave again.

A vertex with 5 edges will have the same problem.



So will a vertex with just one edge.

In fact any vertex with an odd number of edges coming from it will have to be an end or a beginning of our path. But any vertex with an even number of edges won't have that problem.

The number of edges coming from a vertex is called the **degree** of the vertex. A vertex with an odd number of edges coming from it is called an **odd vertex**; one with an even number of edges is called an **even vertex**. So odd vertices have degree 1, 3, 5, etc. Even vertices have degree 0, 2, 4, 6, etc.

A path can only have one beginning and one end, so if there are only 2 odd vertices, then an Eulerian path is possible, starting at one of the odd vertices and finishing at the other one. If there are no odd vertices, then an Eulerian path is possible and it can start and finish at any vertex we like. If there are more than 2 odd vertices, then the graph is not traversable.

So a graph is traversable if it has no more than 2 odd vertices.

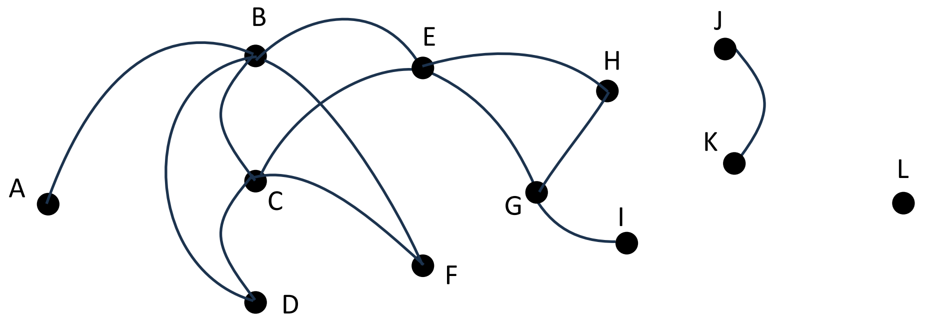
Go back and use this method to check your answers to Q1.

Also check whether the Konigsberg Bridges Problem is possible.

Another Graph/Network

Networks can be used to represent various situations.

Consider 12 people (we'll call them A, B, C etc.) who work at a McDonalds store. Some of them are friends, some aren't. We can represent the workers as vertices on a graph and the friendships as edges joining the vertices. The graph might look like this.



Practice

Q2. (a) How many friends does E have in the store?

(b) How many does H have?

(c) How many does A have?

(d) How many does L have?

(e) Who has the most friends?

(f) Who has the least?

(g) Which people have just one friend?

Q3. Sometimes people need to pass buns to other workers. But the people there are a bit weird and will only pass buns to their friends. How many times will a bun need to be passed to get it

(a) from A to C?

(b) from A to B?

(c) from D to E?

(d) from F to I?

(e) from E to K?

(f) What is the largest number of times a bun can be passed to get it from H to A without the same person handling it twice?

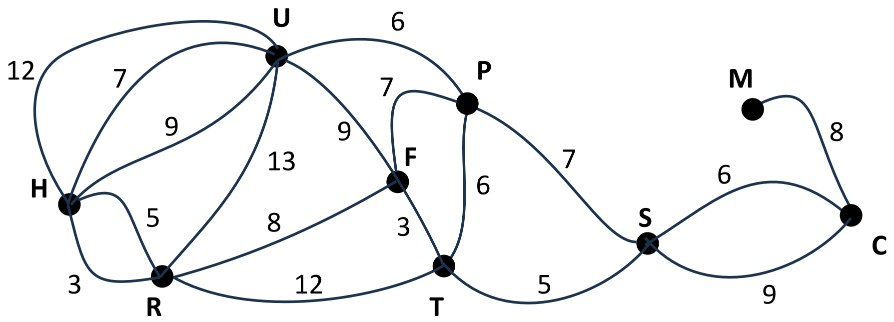
(g) How about from B to F?

Q4. D decides to have a party. She invites all her friends and tells them they can bring their friends. How many people from the store can go to the party (including D)?

Q5. B leaves the store to concentrate on his modelling work for a sports magazine. He is replaced by M who is friends with A, G, I and L. Draw the new graph.

Weighted Graphs

This graph shows the eight towns, Humerus (H), Radius (R), Ulnar (U), Femur (F), Tibia (T), Patela (P), Sacrum (S), Cranium (C) and Mandible (M) and on Boney Island and the roads connecting them. It also shows the lengths of the roads in kilometres. Having numbers on the edges makes it a weighted graph.



Practice

Q6. Nobby wants to drive from Humerus to Ulnar.

(a) How many roads go directly there without going through another town?

(b) How long is the shortest route?

He then has to drive to Radius.

(c) How long is the shortest route from Ulnar to Radius?

Q7. If Nobby wanted to take the longest route possible from Ulnar to Radius without backtracking or passing through the same town twice, what route should he take? (Write it as a sequence of letters like UPTR.) How long would the journey be?

Q8. Is it possible to do a tour of all the roads on the island without driving the same road twice? Give such a route or explain why it's not possible?

Q9. Is it possible to do a tour of all the roads on the island except the road from Cranium to Mandible without driving the same road twice? Give such a route or explain why it's not possible?

Q10. What route should one take to get from Humerus to Mandible driving the least distance? How long would the route be?

Images: rawpixel.com

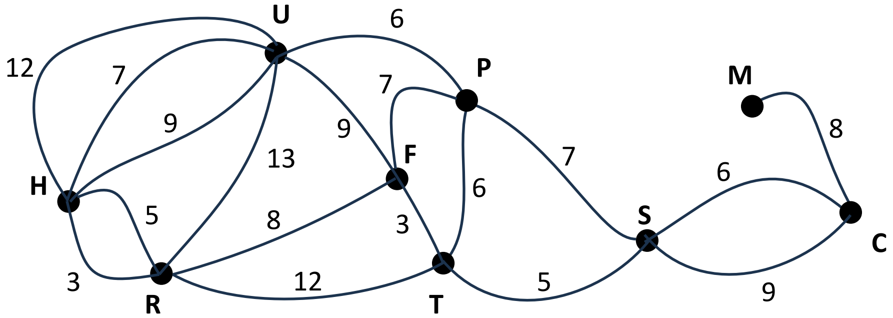
Shortest Path Algorithm

In Q10 you had to find the shortest path from Humerus to Mandible. Checking a few routes would do the job.

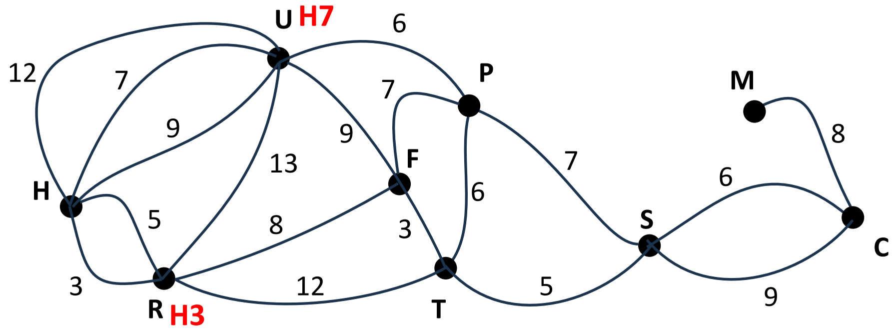
But, if there were a lot more towns and a lot more roads, this would be a more difficult task and it would be easy to miss some routes.

However, there is an algorithm that makes it easy and systematic, so you don't risk missing any routes. It's called the **Shortest Path Algorithm**. An algorithm is a series of simple steps which, if carried out in order, will solve a more complex problem.

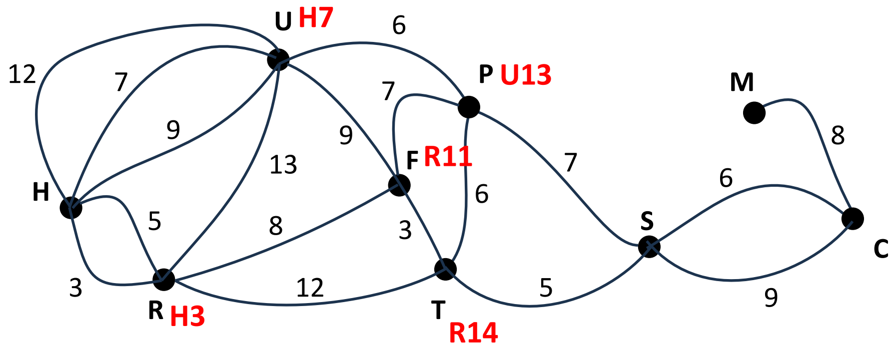
We will use the shortest path algorithm to find the shortest route from Humerus to Mandible.



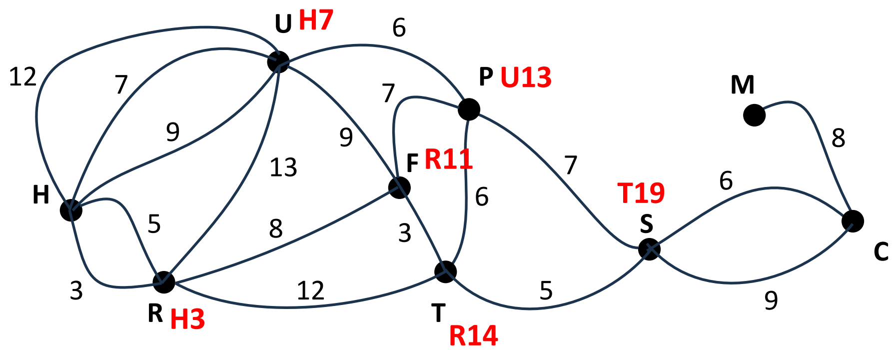
For each town which is connected directly to H (i.e. U and R), we write the shortest distance from H and the letter H.



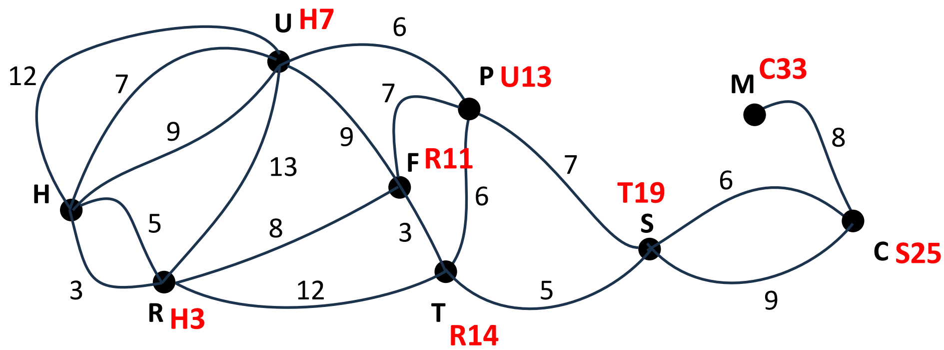
Then for each new town which is connected directly to U or R (i.e. P, F and T, but not H because we have already been there), we write the cumulative shortest distance from H and the letter U or R depending on whether the route comes via U or R.



Then we do the same again. This time the only new town connected to these is S, so we do the same for S.



Then likewise for C and then M



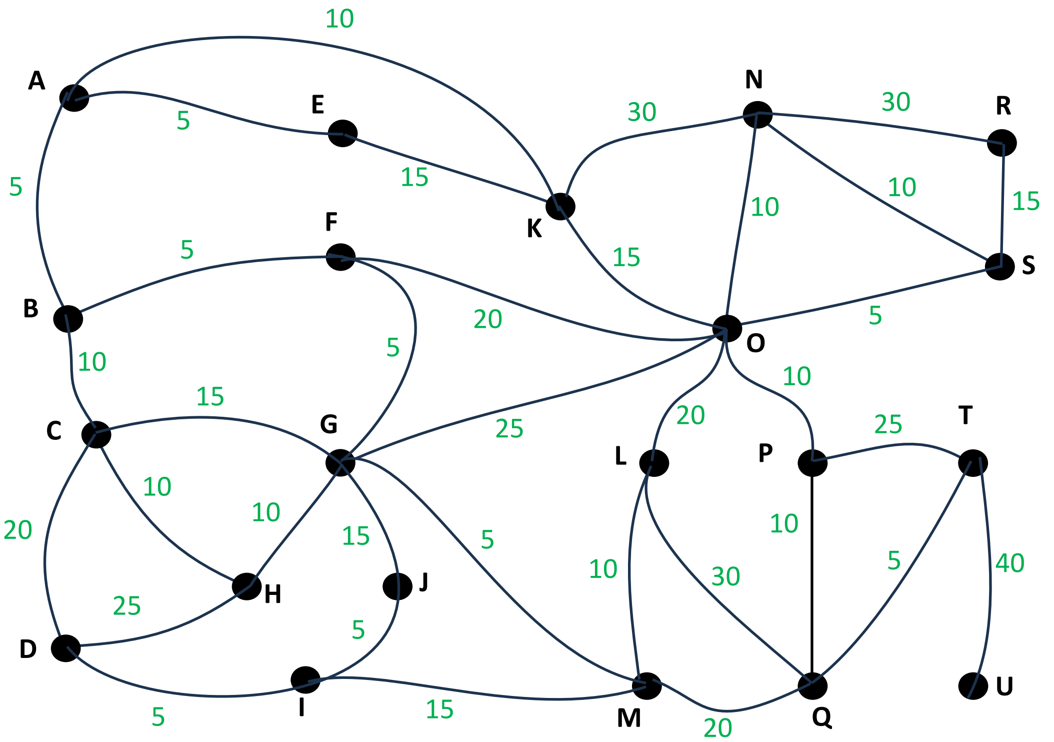
We now know that the shortest path is 33 km long.

To find the route it takes, we trace back from M using the letter written beside it to know which town to go back to next. C is written beside M, so we go back to C, then S is written there, so we go back to S and so on. We write these down in reverse order (each new letter written to the left of the previous one) like this:

HRTSM

Practice

Q11. This is a diagram of a network of Internet servers. The vertices are servers and the edges are connections between them. The time in milliseconds for a message to pass from one server to another is shown by the numbers on the edges.



Use the shortest path algorithm to find:

(a) Which route is the fastest from B to L and how long it takes,

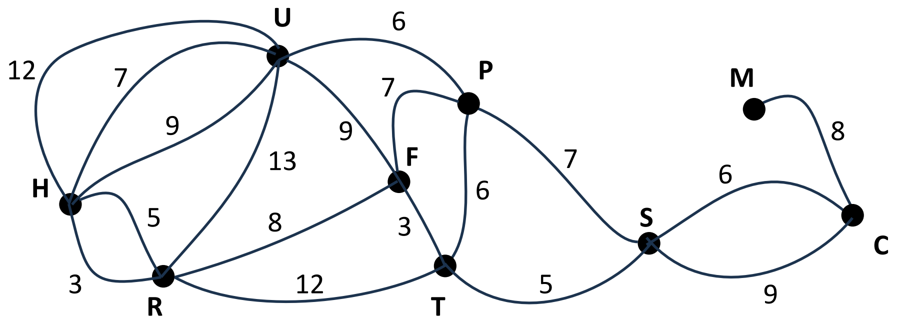
(b) Which route is the fastest from A to U and how long it takes,

(c) Which route is the fastest from D to S and how long it takes.

Note: Vertices on graphs are sometimes called nodes and edges are sometimes called arcs. Just be aware if you read any other texts on graphs.

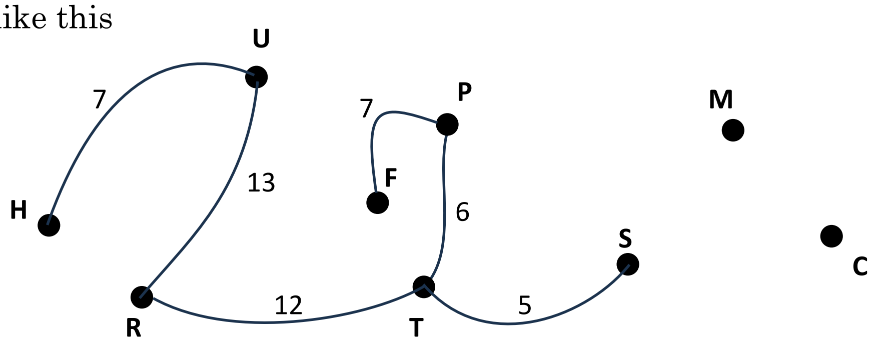
Solve

Q51 **Minimum Spanning Tree**

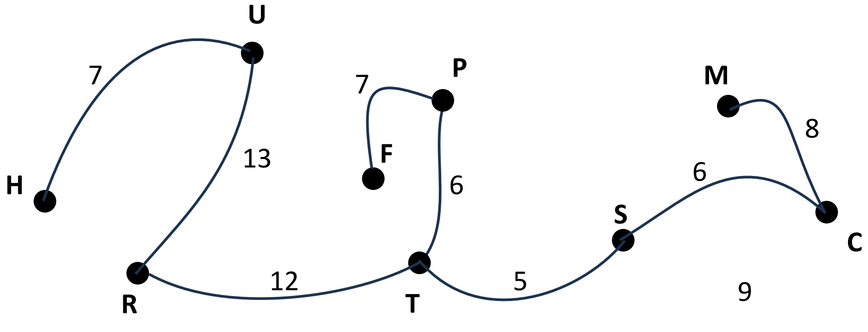


The network above is the towns and roads on Boney Island that we saw earlier. Imagine that a gas plant is installed in the town of Tibia (T). We need to run gas pipes to all the other towns. How should we lay the pipes so as to use the shortest possible length of pipe? We don't need to run pipes along all the edges: if we have a pipe from T to F and one from F to P, we don't need one along the edge from T to P.

A tree is a set of edges all connected by vertices and without any loops. It might look like this



A spanning tree is a tree that connects all the vertices. It might look like this.

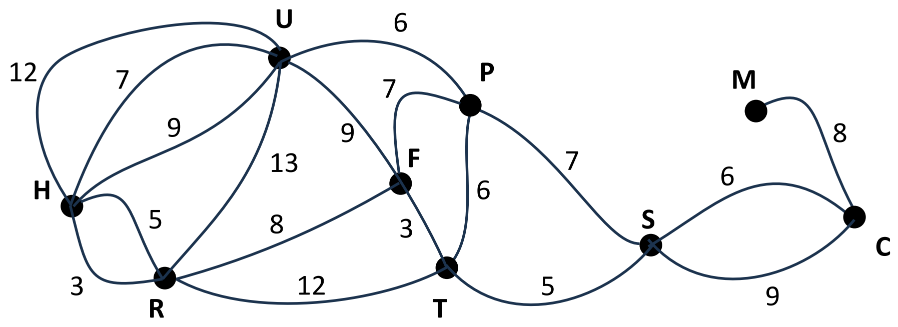


The minimum spanning tree is the spanning tree with the smallest total length of edges. The most economical way to connect gas to all the towns is by a minimum spanning tree. So what we need to find is the minimum spanning tree. Draw the vertices and the minimum spanning tree.

There is a simple algorithm for finding a minimum spanning tree that doesn't rely on trial and error. You might like to see if you can work it out or Google it.

Q52 **Travelling Salesperson Problem**

Let's have another look at the towns and roads on Boney Island.



This time, we have a salesperson who lives at T and needs to visit each town on her daily round and end up back home at T. What route should she take and how far will she drive?

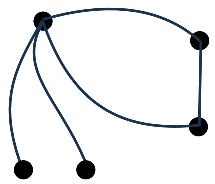
For large networks, the travelling salesperson problem is a difficult one requiring a complex algorithm, a sophisticated computer program and a long running time. You might like to Google it (or you might not). You will more likely find it under 'travelling salesman problem'. We've used 'salesperson' because women can sell things too.

Revise

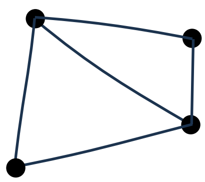
Revision Set 1

Q61. For each of the graphs below, decide whether it is traversable, i.e. if it has an Eulerian path.

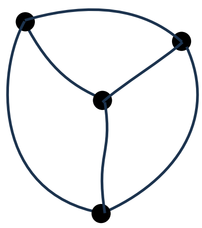
(a)



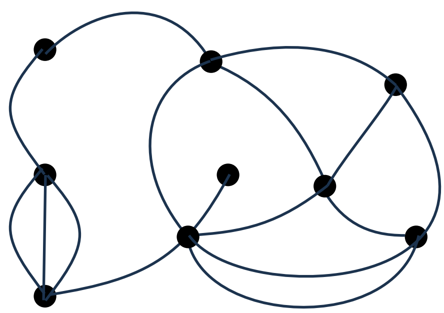
(b)



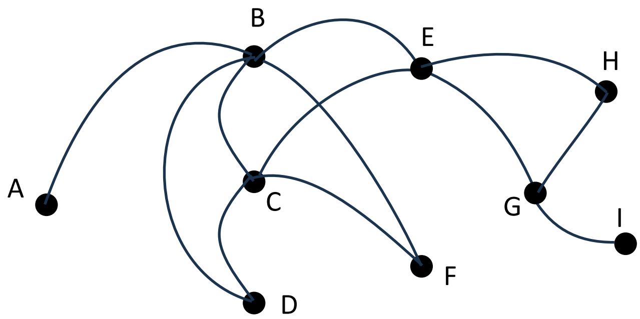
(c)



(d)



Q62. This network shows which football teams have played each other this season.



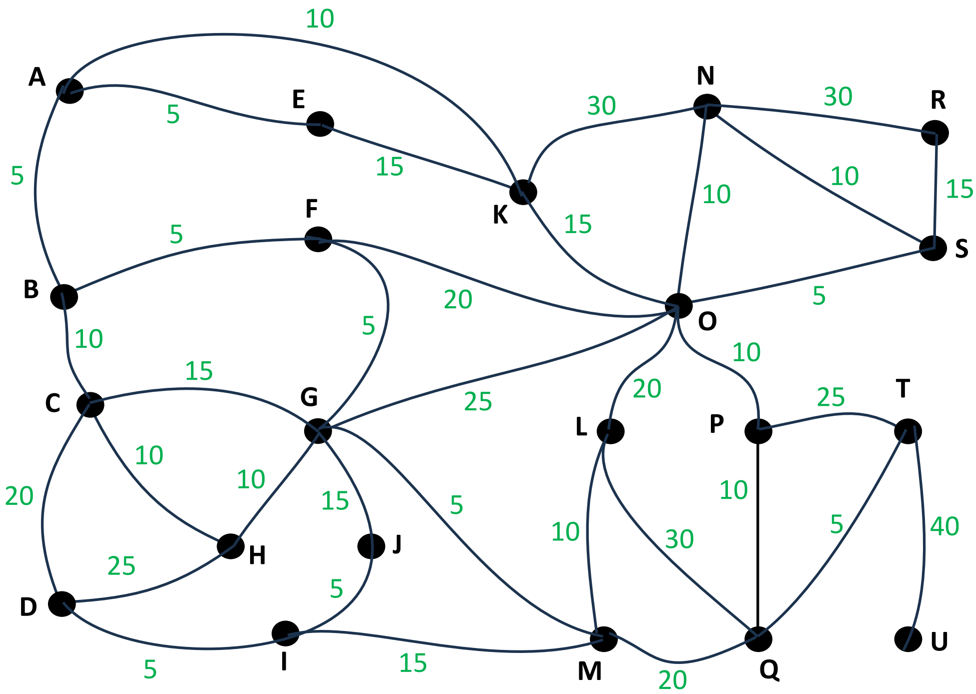
(a) How many games have been played? (Assume no team has played any other team twice.)

(b) Which team has played the most games?

(c) Which teams have played team C?

(d) Which teams have played teams that have played team H?

Q63. This is the network of Internet servers we met earlier.



Assuming that data is sent along the fastest route, how long would it take to go

(a) from A to E

(b) from A to K

(c) from A to G

(d) from A to Q

Answers

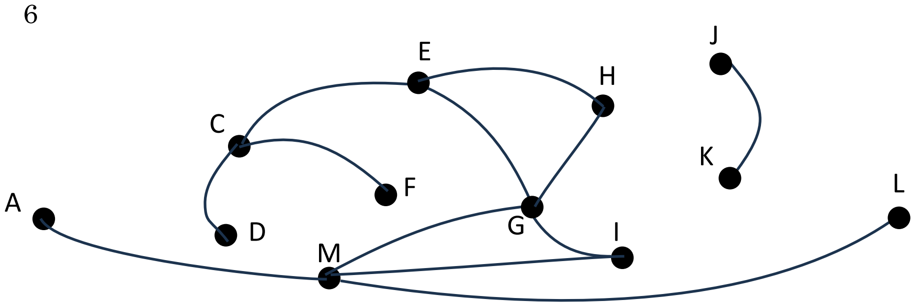
Q1 (a) yes (b) yes (c) no (d) yes (e) no (f) no
(g) yes

Q2 (a) 4 (b) 2 (c) 1 (d) 0 (e) B (f) L
(g) A, I, J and K

Q3 (a) 2 (b) 1 (c) 2 (d) 4 (e) can't be done
(f) 6 (g) 3

Q4. 6

Q5.



Q6. (a) 3 (b) 7 km (c) 12 km

Q7. UFPSTR 40 km

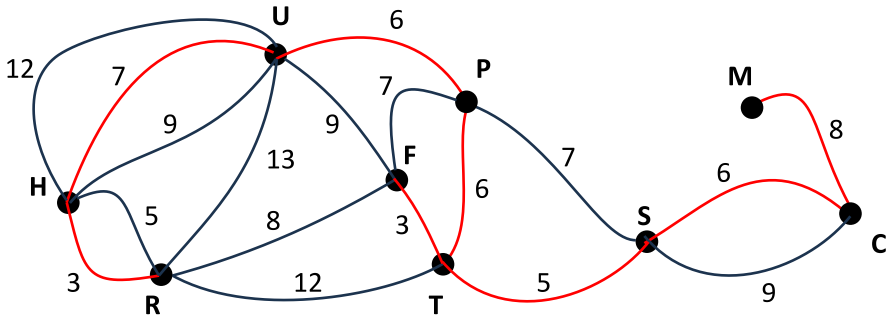
Q8. Not possible because H, R, C and M are odd vertices

Q9. Many routes possible, all starting and ending at H and R, e.g. HUHRHUPSCSTPFTRFUR

Q10. HRFTSCM, 33 km

Q11. (a) BFGML 25 ms (b) ABFGMQTU, 85 ms, (c) DHGOS or DIMLOS, both 50 ms

Q51. Minimum spanning tree in red



Q52. More than one possible. One is TSCMCSPUHRFT, 67 km

Q61. (a) yes (b) yes (c) no (d) yes

Q62. (a) 12 (b) B (c) B, E, D, F (d) H, I, G, E, B, C

Q63. (a) 5 ms (b) 10 ms (c) 15 ms (d) 40 ms