Graph Theory: How Google Maps Plans Routes

Key Stage 4 Programme

Pupil Name

Handbook Designed by A. McGaw

Timetable and Assignment Submission

Timetable – Tutorials

Tutorial	Date	Time	Location
1	12th December 2018	N/A	N/A
2	20th December 2018	3:00pm	
3	10th January 2019	3:00pm	
4	17th January 2019	3:00pm	
5	24th January 2019	3:00pm	
6 (Feedback)	31st January 2019	3:00pm	
7 (Feedback)	7th February 2019	3:00pm	

Timetable – Homework Assignments

Homework Assignment	Description	Due Date
Tutorial 1	Baseline Assessment	18th December 2018
Tutorial 2	Algorithms	8th January 2019
Tutorial 3	Walks, Trails, and Paths	15th January 2019
Tutorial 4	Directed Graphs	22nd January 2019
Tutorial 5	Draft Final Assignment	29th January 2019

Assignment Submission – Lateness and Plagiarism

Lateness	
Submission after midnight	10 marks deducted
Plagiarism	
Some plagiarism	10 marks deducted
Moderate plagiarism	20 marks deducted
Extreme plagiarism	Automatic fail

Uni Pathways – Pupil Feedback Report

Grade	Marks	Guidance
1st	70+	
2:1	60-69	
2:2	50-59	
3rd	40-49	
Working towards a pass	0-39	
Did not submit	DNS	No assignment received by The Brilliant Club

Lateness	
Any lateness	10 marks deducted
Plagiarism	
Some plagiarism	10 marks deducted
Moderate plagiarism	20 marks deducted
Extreme plagiarism	Automatic fail

Name of PhD Tutor	A. McGaw		
Title of Assignment	Plan the Best Route		
Name of Pupil			
Name of School	Essa Academy		
ORIGINAL MARK / 100		FINAL MARK / 100	
DEDUCTED MARKS		FINAL GRADE	

If marks have been deducted (e.g. late submission, plagiarism) the PhD tutor should give an explanation in this section:

Learning Feedback Comment 1 - *Enter Key Learning Priority Here*	
What you did in relation to this Key Learning Priority	How you could improve in the future
Learning Feedback Comment 2 – *Enter Key Learning Priority Here*	
What you did in relation to this Key Learning Priority	How you could improve in the future
Learning Feedback Comment 3 – *Enter Key Learning Priority Here*	
What you did in relation to this Key Learning Priority	How you could improve in the future
Resilience Comment	
How you showed learning resilience during the course	How you could build learning resilience in the future

Contents

Course Rationale	P5
Mark Scheme	P6
Glossary of Keywords	P8
Knowledge Organiser	P10
Tutorial 1	P11
Tutorial 2	P16
Tutorial 3	P21
Tutorial 4	P28
Tutorial 5	P34
Final Assignment	P38
Tutorial 6	P42
Tutorial 7	P43
Appendices	P45

Course Rationale

Whether you're using your GPS to navigate home or chatting with your friends on the internet, graphs underpin almost every aspect of our connected society. With them, your GPS can calculate the fastest route, and your computer knows how to connect to a vast network of billions of machines. But how? This course is designed to push mathematics enthusiasts to answer this question – even though computers can use algorithms perform these incredible tasks, they were all designed by people like you.

There's a joke that all mathematicians are lazy. In fact, mathematicians are always thinking – trying to apply their abstract knowledge to real world problems. In the first tutorial, we'll learn how a so-called lazy mathematician thought about a simple problem long enough that he eventually invented an entirely new branch of mathematics. Graph theory is now a powerful subject with applications from train schedules to brain surgery.

From there we'll delve deeper into the subject. In tutorial two we will study special types of graphs called trees, and in tutorial three learn an algorithm we can use to find them. Finally, we will study Dijkstra's algorithm for finding the quickest route across a graph. Variations of this algorithm are still used today in modern map software and SatNavs.

Throughout every tutorial we will discuss the many practical applications of all the theory we learn. Each homework assignment will introduce a new application of graph theory, as well as ask you to research a new topic which will serve as an introduction for the next tutorial. In the final assignment, you will demonstrate that you can execute all the algorithms we have learned in a real-world context, and show that you can learn and apply an algorithm all by yourself.

Mark Scheme

STEM Mark Scheme

Skills	1st (70-100)	2:1 (60-69)	2:2 (50-59)
Knowledge and Understanding	• All content included is relevant to the general topic and to the specific question/title • Good understanding of all the relevant topics. • Scientific terms are defined and used accurately throughout • Clear justification on how the content included is related to the specific issues that are the focus of the assignment	• Most of the content included is relevant to the general topic and to the specific question/title • Good understanding of most the relevant topics • Scientific terms are used accurately but not always clearly defined. • Adequate justification on how the content included is related to the specific issues that are the focus of the assignment	• Some of the content included is relevant to the general topic and to the specific question/title • Good understanding on some of the relevant topics but occasional confusion on others. • Scientific terms are used mostly accurately with occasional confusion and often not defined. • Some justification on how the content included is related to the specific issues that are the focus of the assignment
Critical Evaluation / Problem solving	• Moved beyond description to an assessment of the value or significance of what is described • Evaluative points are consistently explicit/systematic/reasoned/justified • Effective critiques on the reliability of sources provided • Consistently demonstrate clear, analytical and logical steps to solving problems	• Mostly description but some assessment of the value or significance of what is described • Evaluative points are mostly explicit/systematic/reasoned/justified • Some evidence of critiques on the reliability of sources provided • Some examples of solving problems but not consistently clear, analytical and logical	• Only description with minimal assessment of the value or significance of what is described • Evaluative points are at times explicit/systematic/reasoned/justified • Limited evidence of critiques on the reliability of sources provided • Demonstrate clear steps to solving problems but not consistently analytical or logical

Structure and Presentation	• Ideas are presented in paragraphs and arranged in a logical structure that is appropriate for the assignment • The introduction clearly outlines how the essay/report will deal with the issues • The conclusion summarises all the main points clearly and concisely • All calculations, formulas and methods are clearly structured, clear to follow and correct • Tables and graphs are effectively constructed including appropriate headings, units and scales. • All sources are referenced correctly in an agreed format	• Ideas are presented in paragraphs and arranged in a structure that is mostly appropriate for the assignment • The introduction adequately describes how the essay/report will deal with the issues • The conclusion summarises most of the main points clearly • Calculations, formulas and methods are mostly structured, clear to follow and correct • Most tables and graphs are well constructed • Most sources are referenced correctly in an agreed format	• Ideas are presented in paragraphs and arranged in a structure • The introduction mentions how the essay/report will deal with the issues • The conclusion summarises some of the main points clearly • Calculations, formulas and methods are not always structured, clear to follow and correct. • Some tables and graphs are well constructed but contains some errors • Some sources are referenced correctly in the agreed format with occasional errors

Subject Vocabulary

Word	Definition	In a sentence
Algorithm	A step-by-step sequence of operations which solves a problem	We will apply an algorithm to solve the problem.
Arc	Another word for edge; a line joining two vertices.	An arc connects vertex A to vertex B.
Connected	Two vertices are connected if there is a walk from one vertex to the other.	A and D are connected as A-B-C-D is a walk from A to D.
Connected Graph	A graph is connected if every pair of vertices are connected.	This graph is connected.
Cycle	A path that ends where it begins.	A-B-C-D-A is a cycle starting and finishing at A.
Degree	The number of edges joining to a vertex.	The degree of this vertex is 4 because there are 4 edges connected to it.
Digraph	Also called a directed graph, a digraph is a graph where all its edges have a direction.	This digraph represents a one-way system of roads.
Dijkstra's Algorithm	An algorithm used to find the shortest path between two vertices on a weighted graph.	Let's apply Dijkstra's algorithm to solve this problem.
Edge	A line joining two vertices.	Vertices A and B are joined by an edge.
Graph	A mathematical structure composed of vertices joined together by edges.	In this course, we will study graphs.
Handshake Theorem	A statement by Euler which says that the number of edges in a graph is equal to half of its total degree.	We can use the Handshake Theorem to save us counting the edges.
Integer	Another word for a whole number. ..., -3, -2, -1, 0, 1, 2, 3, ... are all integers.	1 and 2 are integers.
Isomorphic	Two graphs are isomorphic if the vertices of one can be shifted around until the graph looks identical to the other.	By rearranging vertices, we see that these graphs are isomorphic.

Kruskal's Algorithm	An algorithm used to find the minimum spanning tree of a weighted graph.	Let's apply Kruskal's Algorithm to find a minimum spanning tree.
Minimum Spanning Tree	A spanning tree of a graph with the smallest possible weight.	Seeing a Minimum Spanning Tree would help to solve this problem.
Multigraph	A graph containing at least one multiple edge – two edges which that connect the same pair of vertices.	There are two edges connecting vertices A and B, so it is a multigraph.
Node	Another name for a vertex; an end point of an edge.	This node has degree 4.
Path	A walk in which no vertices are repeated.	A-B-C is a path because no vertices are repeated.
Spanning Tree	A subgraph which is a tree and contains every vertex	This subgraph is a spanning tree.
Subgraph	A smaller graph which contains a subset of the edges and vertices of the original graph.	This smaller graph is a subgraph of the larger graph.
Total Degree	The total sum of the degrees of every vertex.	We add up the degrees of all the vertices to find the graph's total degree.
Trail	A walk in which no edges are repeated.	A-B-C-B-E is a trail as no edges are repeated.
Tree	A connected graph which contains no cycles.	This graph is a tree.
Vertex (plural: Vertices)	An end point of an edge.	This vertex has degree 3.
Walk	A sequence of edges such that the end point of one edge is equal to the start point of the next edge.	A-B-C-D-A-B-A is a walk.
Weight	A number assigned to an edge.	The weights on this graph represent distance.
Weighted Graph	A graph in which every edge has a weight.	This weighted graph represents a map of a country.

Knowledge Organiser

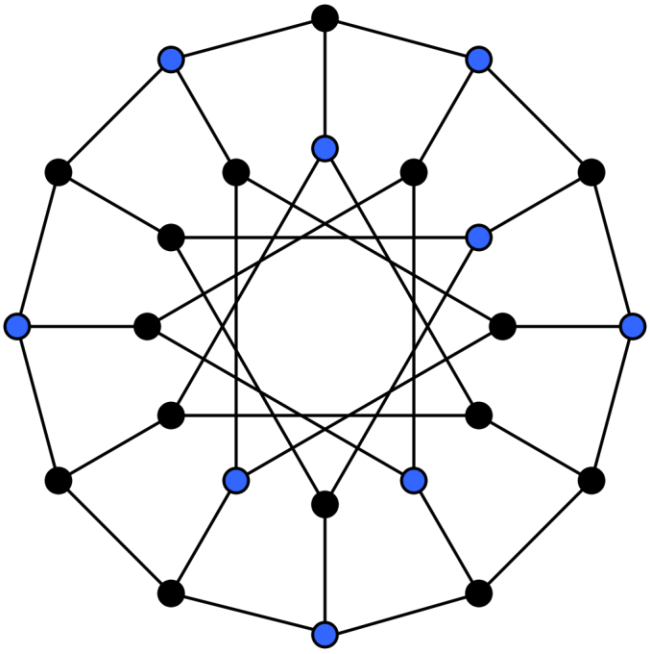
A graph is...	when...	and this means...
Eulerian	all vertices have even degree	it can be drawn without taking your pen off the paper, starting anywhere
Semi-Eulerian	two vertices have odd degree and the rest have even degree	it can be drawn without taking your pen off the paper, starting from odd vertices
Non-Eulerian	it is not Eulerian and not Semi-Eulerian	it is impossible to draw without taking your pen off the paper

KRUSKAL'S ALGORITHM	
What does it do?	Takes any weighted connected graph and finds a **minimum spanning tree** of that graph
Which is...	a tree containing every vertex of the original graph which is of minimal total weight
and we execute it by...	1. Delete all edges 2. Add the edge of smallest weight as long as it does not create a cycle 3. Repeat until the graph is connected
And we must remember...	There might be more than one minimum spanning tree, but the total weight of all of these minimum spanning trees will always be the same!

DIJKSTRA'S ALGORITHM	
What does it do?	Finds the **smallest-weight path** from the start vertex to a destination vertex on any connected graph
Which is...	the journey of smallest weight along edges from vertex to vertex beginning with the start vertex and finishing at the destination vertex
and we execute it by...	1. The start vertex begins as the current vertex 2. Find the distance from the current vertex to its neighbours. Update the distance if it is better 3. Look across the whole graph for the vertex with the smallest distance. Choose this as the new current vertex – circle its distance and go to step 2 4. Repeat until the destination vertex is selected as the current vertex
And we must remember...	When applying to a directed graph, we must always travel *along* the arrows!

This...	can be applied in the real world to...
Graph	Maps, power grids, computer networks
Weights	Travel time, distance, voltage, cost
Kruskal's Algorithm	Lowest cost for total connectivity
Dijkstra's Algorithm	Quickest route, shortest route

Tutorial 1 – What is a Graph?

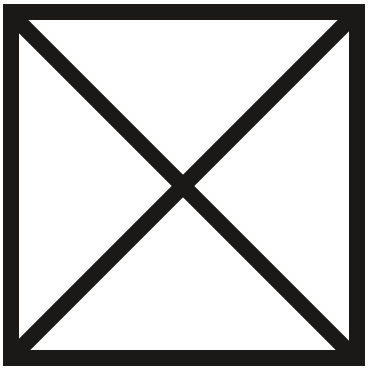


What is the Purpose of Tutorial 1?

- Learn how Leonard Euler first conceived of graphs
- Know the different parts of a graph
- Know how to recognise an Eulerian graph

Can you draw this shape?

Try and draw the shape below without taking your pen off the paper and without retracing your steps. Folding or tearing the paper is also not allowed!

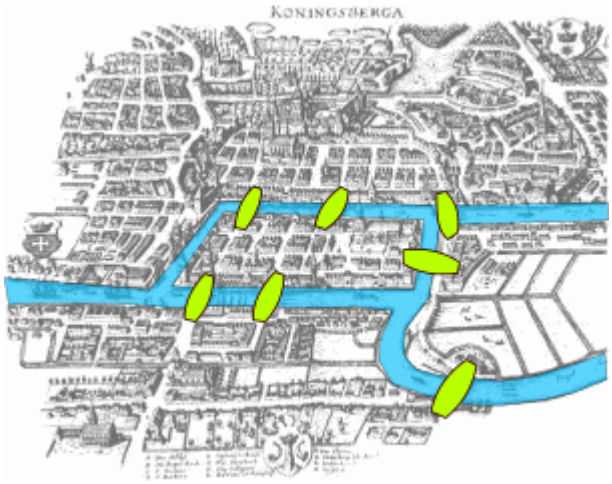


Initial thoughts:

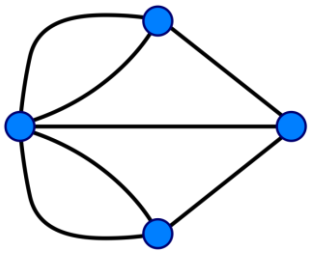
Have you done it, or do you think it's impossible? Why?

The Seven Bridges of Königsberg

Below is an image of Königsberg, a city of Prussia which was destroyed during the Second World War. People wondered whether it was possible to walk through the city in such a way that you crossed every bridge exactly once. Do you think it's possible?

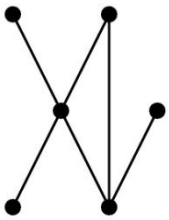


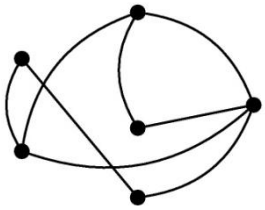
Leonard Euler was the first to solve the problem. He observed that it doesn't matter which route through a certain landmass you take – it only matters which bridges you choose to cross. Because of this, he compacted each landmass into a single point, then drew lines between them representing the bridges. The resulting structure is called a graph:

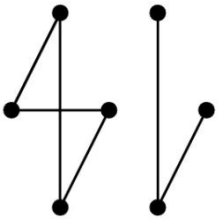


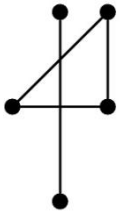
Label the key features of the graph above (vertex, edge, degree).

The following are all examples of graphs.









Notice that the first two graphs are formed of one part – these graphs are called *connected*. The third and fourth graphs are still examples of a single graph, but are *disconnected*.

Eulerian and Semi-Eulerian Graphs

A graph is called **Semi-Eulerian** if:

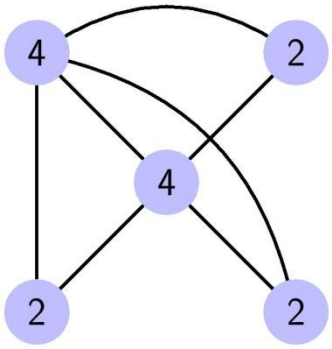
(i) we can draw it without taking our pen off the paper; and

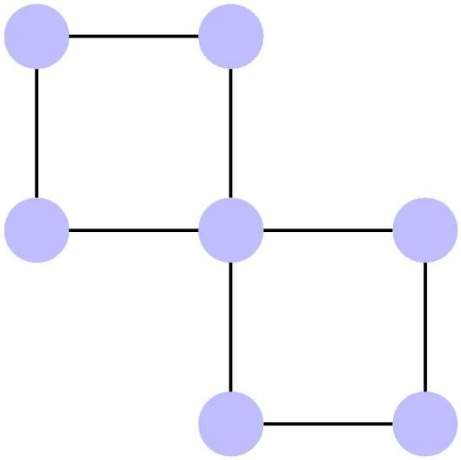
(ii) without retracing our steps.

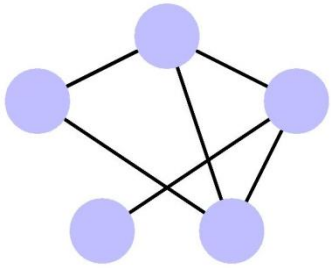
In addition, if we can carry out such a drawing by starting from *any* vertex, the graph is called **Eulerian**. If a graph is not Semi-Eulerian, and not Eulerian, we called it **Non-Eulerian**.

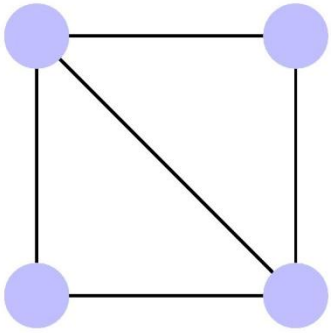
Below are some graphs. Inside each vertex write its degree, then guess whether the graph is Eulerian, Semi-Eulerian, or not Eulerian. For example:

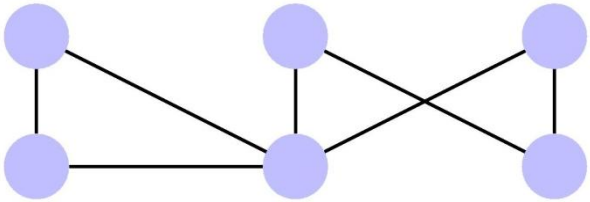
This is Eulerian:











What makes a graph Semi-Eulerian or Eulerian?

Assignment 1 – Baseline Assessment

Complete all homework on a separate sheet of paper.

1. Which of the following numbers are even numbers?

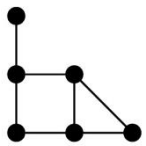
73 144 2 1 16 1,798 625

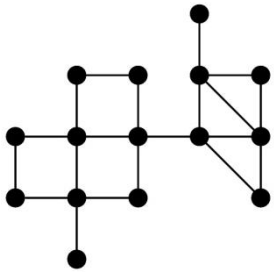
2. In your own words, what is an even number?

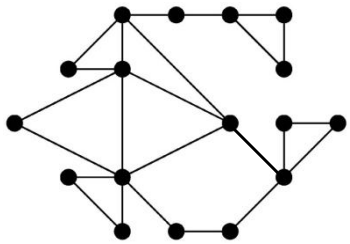
3. Is the following statement true or false: If x is an even number, then we can find a whole number n with the property that $x = 2n$. Why/why not?

4. Is the following statement true or false: If x and y are even numbers, then $x + y$ is an even number. Try to prove why your answer is correct.

5. Without trying to draw them, decide whether these graphs are Eulerian, Semi-Eulerian, or not Eulerian.







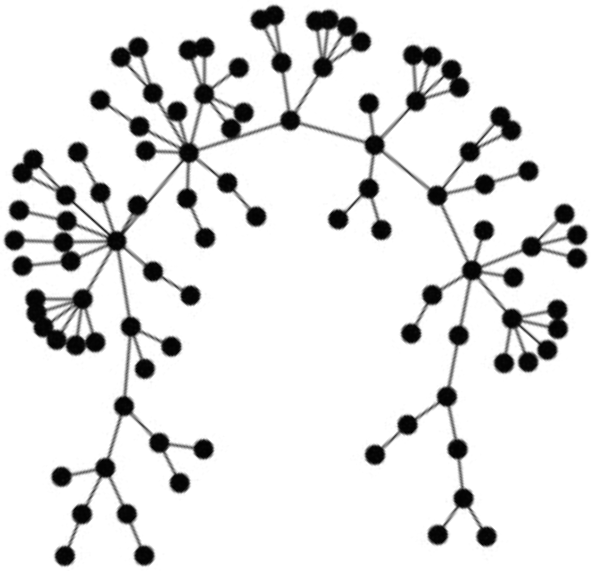
6. For the graphs in question 5 which are *not* Eulerian, how many more edges must you add in order to make them Eulerian?
7. Draw three graphs with 8 vertices: one which is Eulerian, one which is Semi-Eulerian, and one which is Non-Eulerian.

Research Question – Weighted Graphs

Research what a *weight* is in the context of graph theory, and what a *weighted graph* is. Next, research or think of as many practical applications of weighted graphs as you can. Why might you want to use a weighted graph instead of just a graph?

A good answer will be 100-200 words long and will contain diagrams/drawings to support the explanations.

Tutorial 2 – Cycles and Trees



What is the Purpose of Tutorial 2?

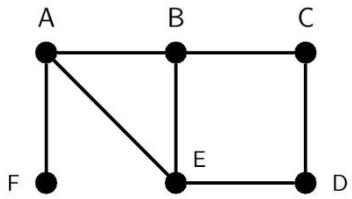
- Learn how to identify cycles on graphs
- Know the definition of a tree and how to identify them

Cycles

A **cycle** is a journey along edges, from vertex to vertex, which:

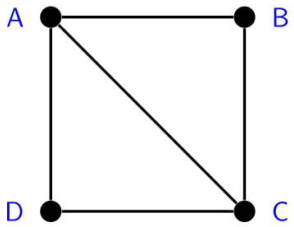
(a) starts and finishes at the same vertex;
(b) does not retrace its steps

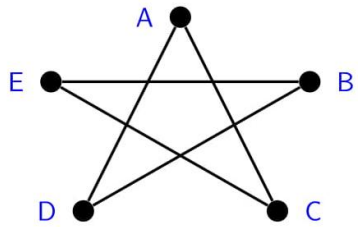
Cycles form "loops" on graphs, but they are not always easy to see!

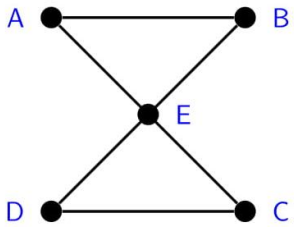


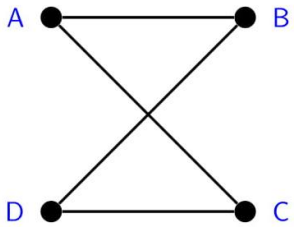
In this graph, A-B-E-A forms a cycle. Because it contains 3 edges, we call it a **3-cycle**. Also, B-C-D-E-B forms another cycle – in this case, a **4-cycle**. There is another cycle – can you spot it? (Hint: cycles can overlap with each other!)

Can you identify the cycles in the following graphs?









Try drawing a graph with 7 vertices which contains a 7-cycle and two 3-cycles:

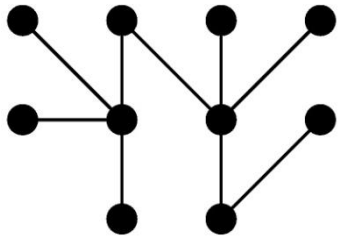
Trees

Now that we know what a cycle is, we can talk about trees. A **tree** is simply any graph which:

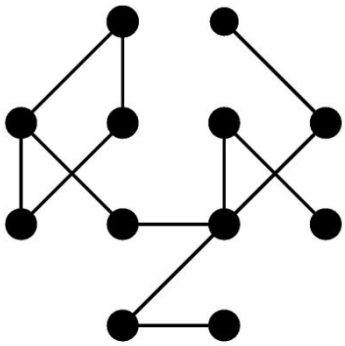
(a) is connected;
(b) contains no cycles.

Try drawing some of your own trees in the space below. As long as your graph satisfies those two conditions, it's a tree!

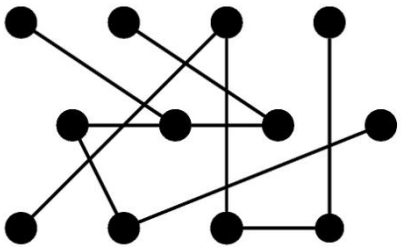
Which of the following graphs are trees? For the graphs which are not trees, give reasons as to why they are not trees.



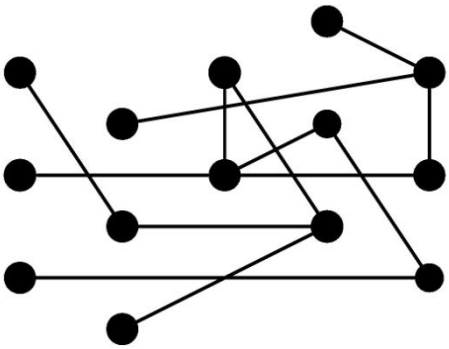
(a)



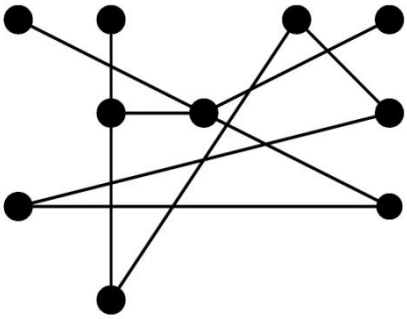
(b)



(c)



(d)

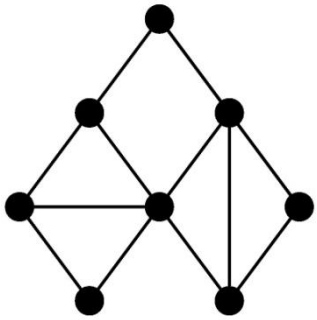


(e)

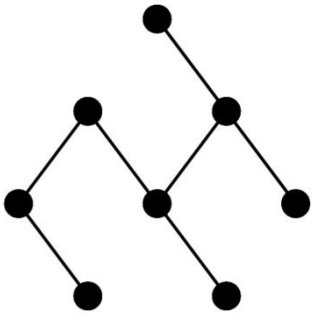
Be careful – cycles aren't always easy to spot! It may help to draw a less complicated isomorphic graph.

Spanning Trees

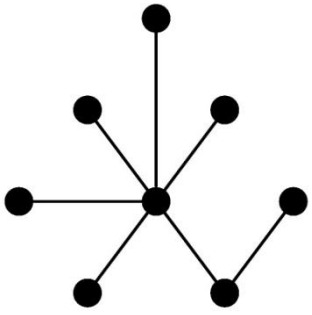
A spanning tree of a graph can be thought of as the "skeleton" of a graph – it is what is left over after we have deleted all of its cycles. A spanning tree must be a tree and must contain all the vertices of the original graph. They are obtained by deleting edges to remove cycles. Consider the graph below:



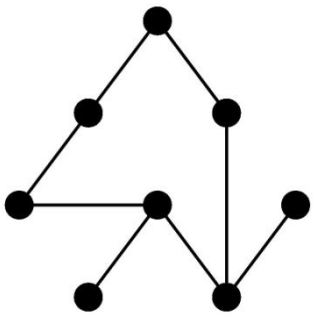
Which of the following are spanning trees of this graph? Why/why not?



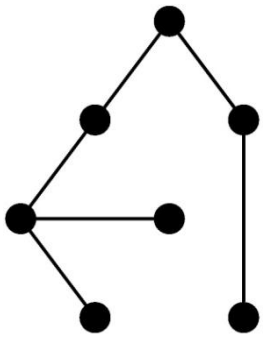
(a)



(b)

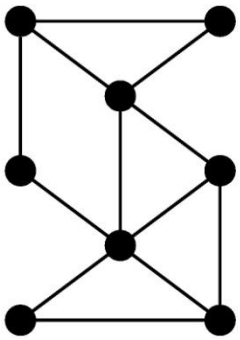


(c)



(d)

Try drawing spanning trees of this graph:

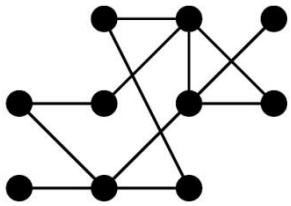


Assignment 2 – Trees and Weighted Graphs

Complete all homework on a separate sheet of paper.

Maths Questions

1. (a) How many cycles are on the graph below?
   (b) Draw five different spanning trees of the graph below.



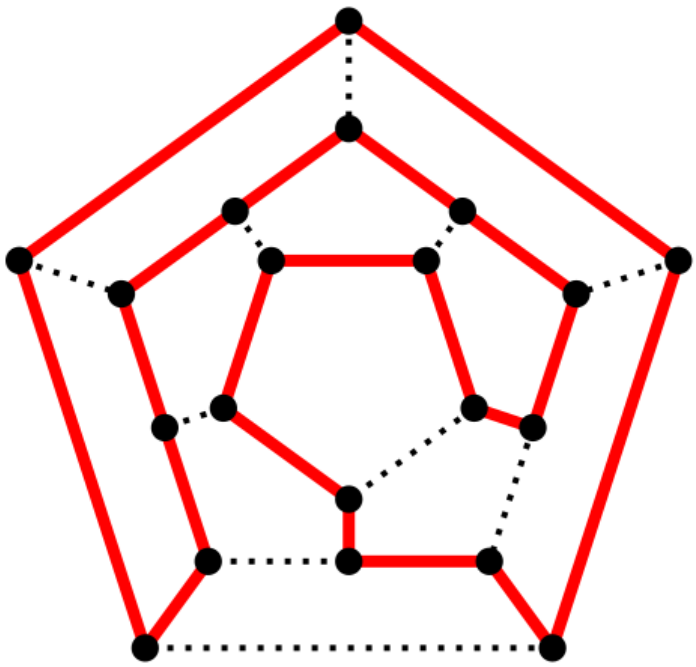
2. (a) Draw a tree with 3 vertices. How many edges does it have?
   (b) Draw a tree with 4 vertices. How many edges does it have?
   (c) Draw a tree with 5 vertices. How many edges does it have?
   (d) Look closely at your last three answers – can you spot a pattern? How many edges would you expect a tree with 10 vertices to have? Draw one and see if you are right?
   (e) Will this pattern always be followed? Is it possible to draw a tree which breaks the pattern?

Research question – algorithms

Research what an *algorithm* is in mathematics and provide some practical uses for them. Next, write a step-by-step algorithm how to make a cup of tea. Be precise as possible! Imagine a machine following your instructions – would it succeed at making a cup of tea?

Aim to write approximately 250 words and include diagrams. Finally, don't forget to give your sources!

Tutorial 3 – Minimum Spanning Trees and Kruskal's Algorithm



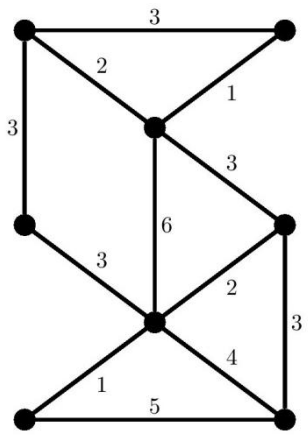
What is the Purpose of Tutorial 3?

- Learn about algorithms and their key features
- Learn about minimum spanning trees and an algorithm used to find them

Minimum Spanning Trees

The first research question asked you to research weighted graphs. Here is some space to jot down what they are as a reminder:

The weight of a graph is the sum of all the weights on its edges. For example, the weight of this graph is 36:



Draw a spanning tree of this graph. What is its weight? A spanning tree which has this smallest possible weight is called a *minimum spanning tree*. But how do we find it?

Kruskal's Algorithm

Now we return to the problem posed earlier: Given a graph, how do we find a minimum spanning tree? One answer is to use *Kruskal's algorithm*, which inputs a graph and outputs a minimum spanning tree:

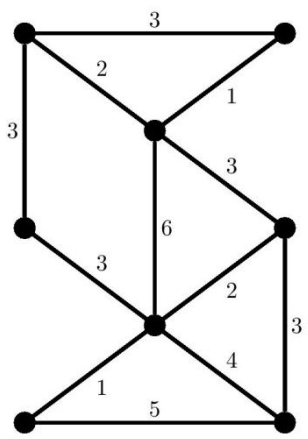
START
INPUT a weighted graph

1. Delete every edge from the graph
2. Add in the edge with the smallest weight (if more than one edge have the smallest weight, add either) as long as adding this edge does not create a cycle
3. If the graph is connected, then STOP. Otherwise, go to step 2.

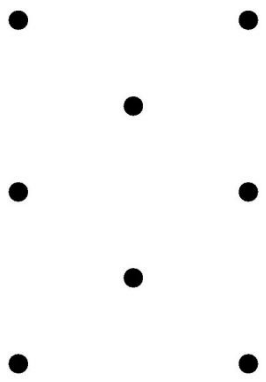
OUTPUT: a minimum spanning tree

Let's try it on the graph we had earlier.

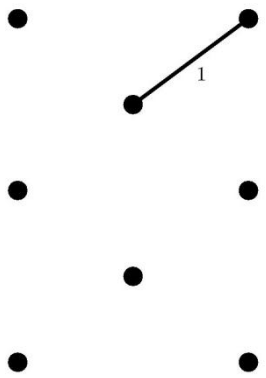
INPUT:



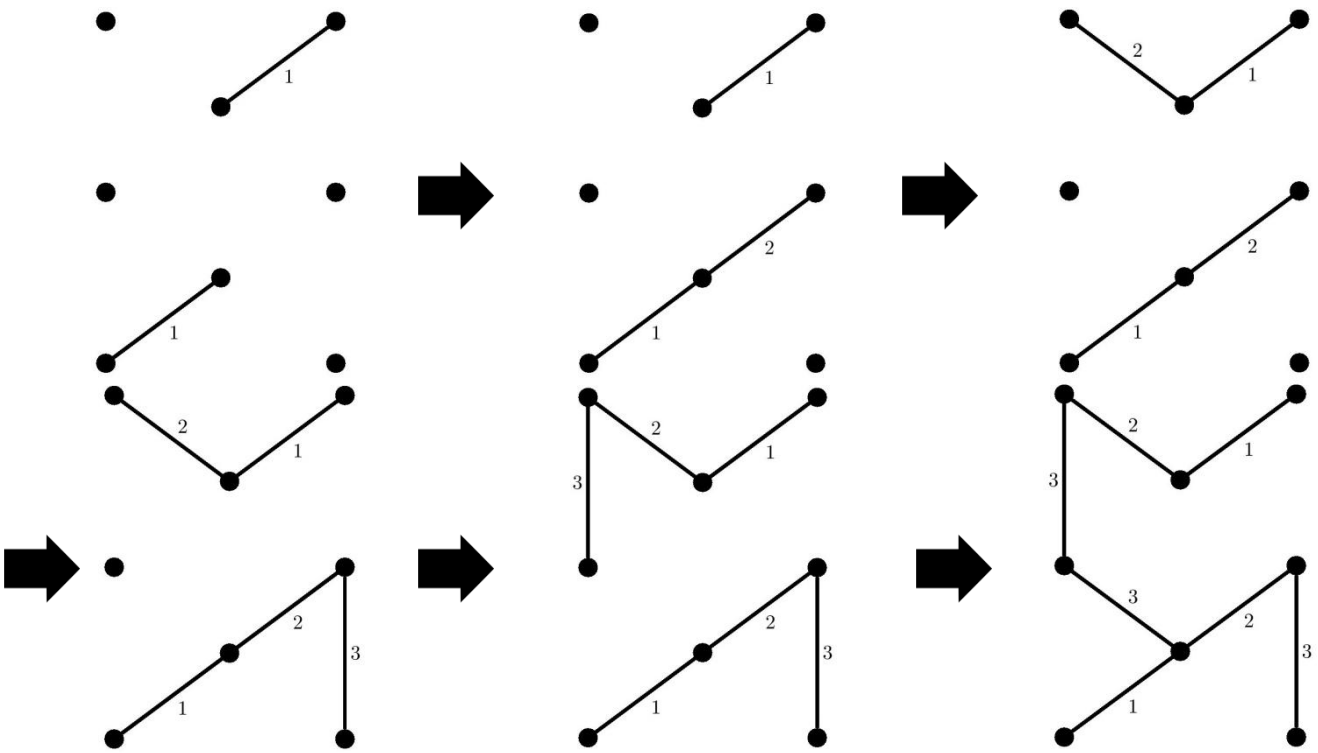
1. Delete every edge from the graph:



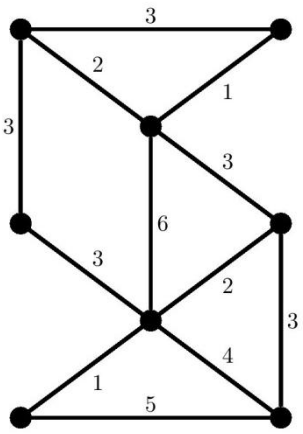
2. There are two edges of smallest weight, which is a weight of 1. We can choose any, provided we don't create a cycle:

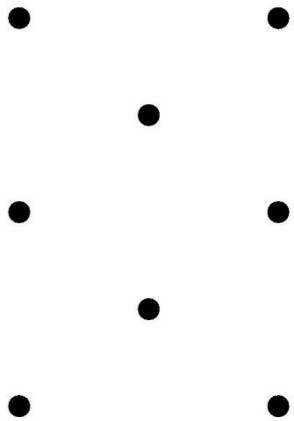


3. Now we look at what we have so far. Is it a tree? No – so we go back to step 2. We do this, adding the smallest weight edge each time, making sure we don't create any cycles.



Is it a tree? Yes, so we STOP. Our output is a minimum spanning tree. Its weight in this case is 15. Note that the minimum spanning tree is not unique – there are other minimum spanning trees for this graph. However, they will all have the same weight. Apply Kruskal's algorithm now and find a **different** minimum spanning tree:





Uses of Minimum Spanning Trees

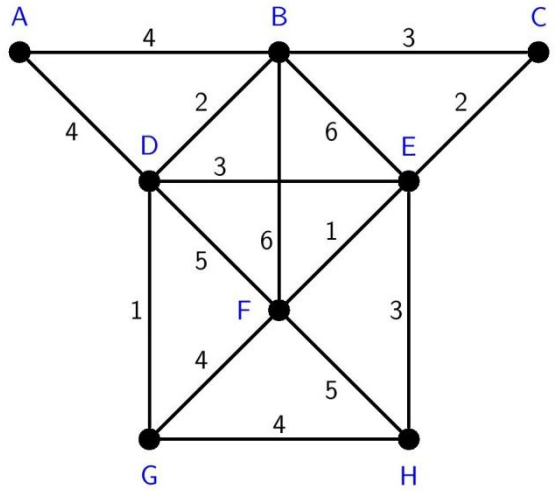
Think about some of the uses for weighted graphs you discovered in your last homework assignment. With those in mind, can you think of any uses for minimum spanning trees?

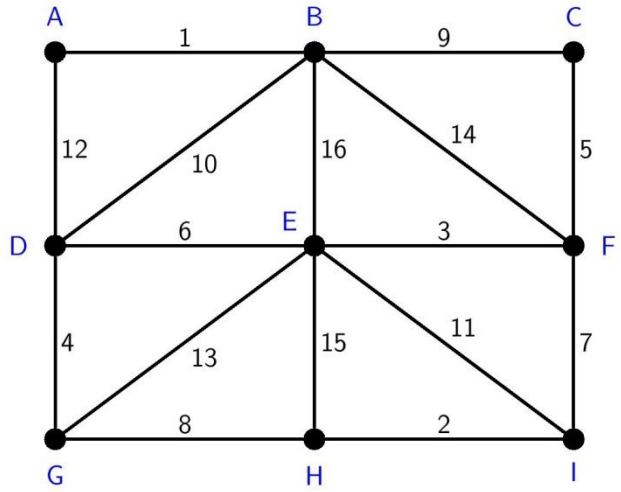
Assignment 3 – Kruskal's Algorithm and Walks, Trails, & Paths

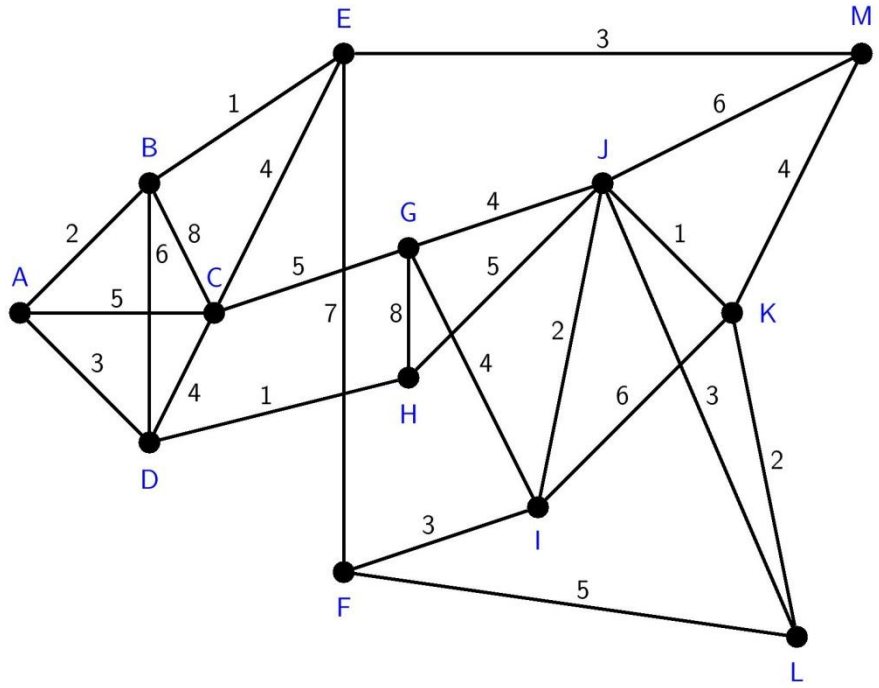
Complete all homework on a separate sheet of paper.

Maths Questions

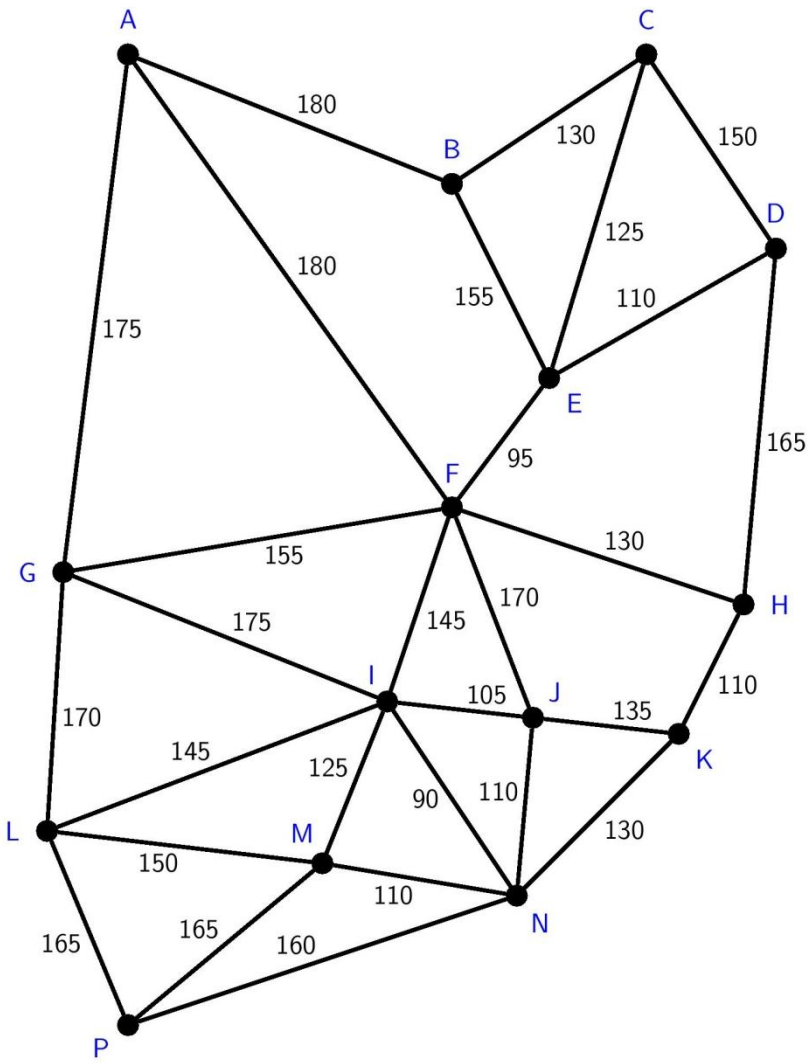
1. Use Kruskal's Algorithm to find minimum spanning trees of these graphs and give their weights. **Important**: When presenting your solutions, list which edges you've added in order.







2. A President wants to install a new railway system across her country to link all the major cities together. Her cabinet presents the following graph. Each vertex is a city and each edge is a railroad connecting two cities. The weights are the costs of building that railroad. The costs are given in thousands of dollars, so a weight of 90 means $90,000, a weight of 165 means $165,000, and so on.

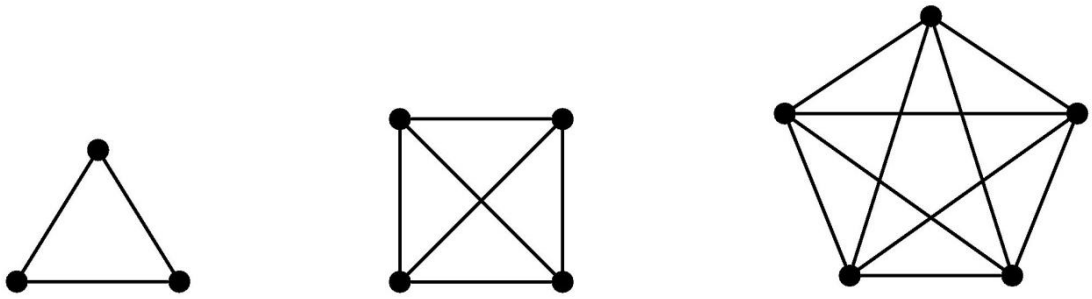


She is unhappy with the graph. The budget for the railway is only $1,850,000, so she cannot afford to build every railroad. She hires you to come up with a solution. Use what you have learned to come up with a way of connecting all the cities together via railroads as cheaply as possible. Can the President afford to install your new railway, and how much will it cost?

Give two disadvantages of your new railway compared to if every railroad were installed.

Can you modify your new railway to fix one of these disadvantages, while still keeping it within the President's budget?

3. The **complete graph** is a graph in which all vertices have an edge between them. Here are the complete graphs with 3, 4, and 5 vertices:



Draw complete graphs with 6 and 7 vertices.

How many edges do these graphs have? Can you think of how many edges the complete graph on *n* vertices will have?
(Hint: Count the edges on the graphs you can draw. Can you spot a pattern?)

Research Question

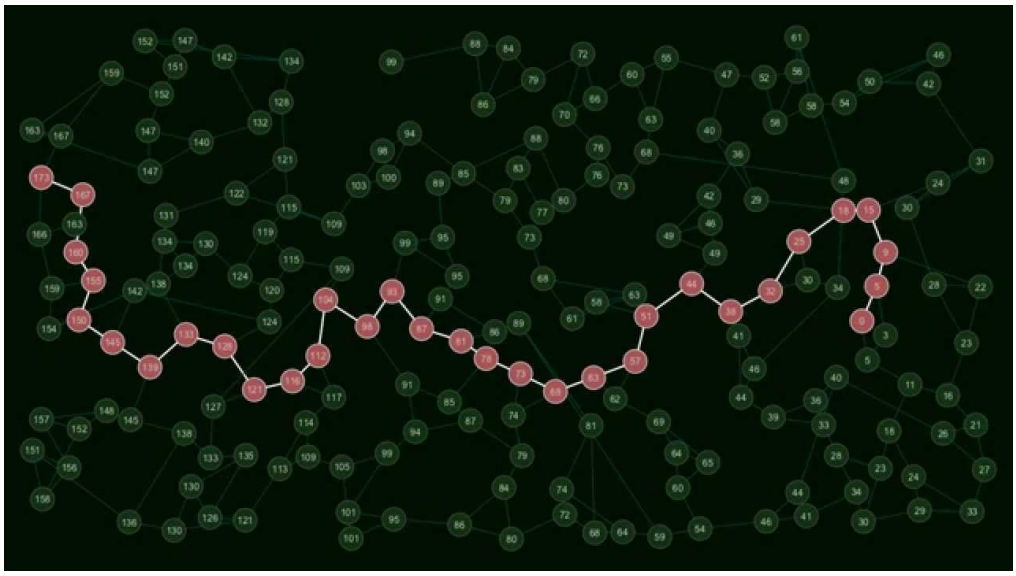
Research the meaning of the following words in the context of graph theory: *walk*, *trail*, and *path*. What are the differences between them?

Next, provide some practical applications for them. What does *distance* mean in the context of graph theory, particularly when looking at weighted graphs?

Finally, draw some graphs which demonstrate the difference between a walk, trail, and path.

Aim for 250 words, include diagrams, and don't forget to give your sources!

Tutorial 4 – Dijkstra's Algorithm



What is the Purpose of Tutorial 4?

- Learn and apply Dijkstra's algorithm for finding the shortest distance between two vertices

Dijkstra's Algorithm

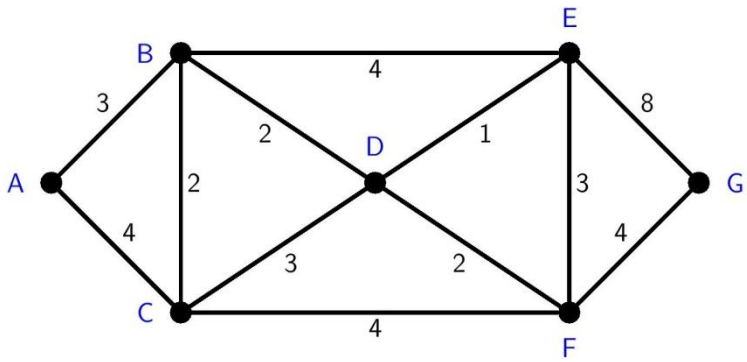
This is one algorithm that GPS systems use to calculate routes. First we will look at the algorithm, then we will see it in action. Again, it looks more complicated in algorithm form. Essentially, we have an origin vertex (where we are travelling form) and a destination vertex. Every vertex will receive a *possible distance*, which is its distance from the origin, but this might change later in the algorithm. Eventually, each vertex will receive a *definite distance* which we will know is the smallest distance to it from the origin.

START
INPUT: A weighted graph with one vertex marked as the *origin* and a different vertex marked as the *destination*.

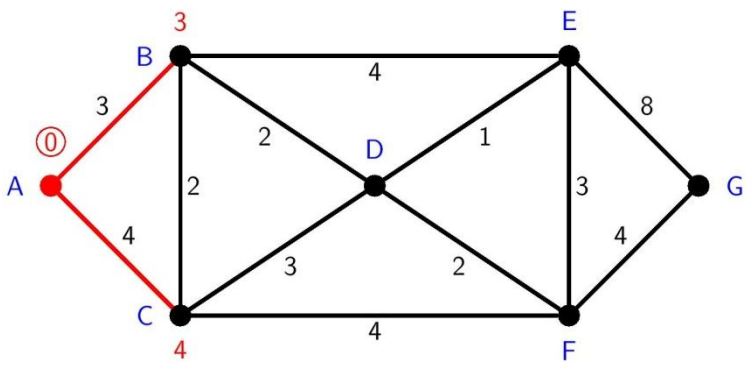
1. Mark all vertices as *unvisited*
2. Set the origin as the *current vertex*, mark it as *visited*, and set its *definite distance* as 0
3. For each neighbouring unvisited vertex, calculate its *possible distance*, which is the current vertex's *definite distance* plus the weight of the edge between them. If this is smaller than that neighbour's current *possible distance*, replace it. (For example, if vertex B has a possible distance of 6 and we calculate a new possible distance of 5, then vertex B's new possible distance will be 5.)
4. Find the smallest *possible distance* among all vertices on the whole graph. The vertex with the smallest *possible distance* becomes the new *current vertex*
5. Mark this current vertex as *visited* and its *possible distance* becomes its *definite distance*.
6. If the current vertex is the destination vertex, STOP. Otherwise, go back to step 3.

OUTPUT: The shortest distance from the origin vertex to the destination vertex. We can also use this to plot the shortest path from the origin to the destination.

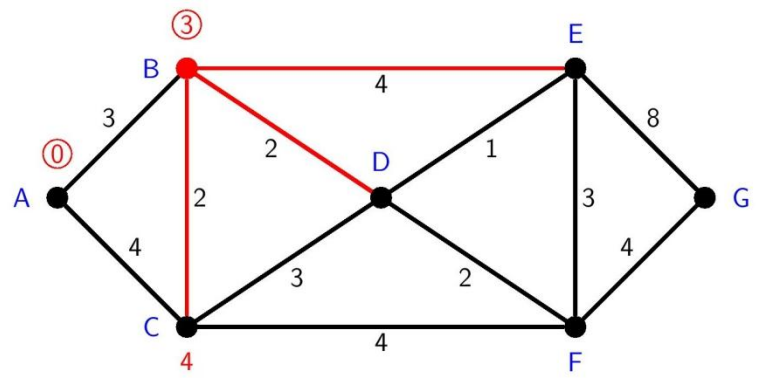
Let's look at the following graph and apply Dijkstra's algorithm to find the shortest path from A to B. The weights are travel times between each vertex.



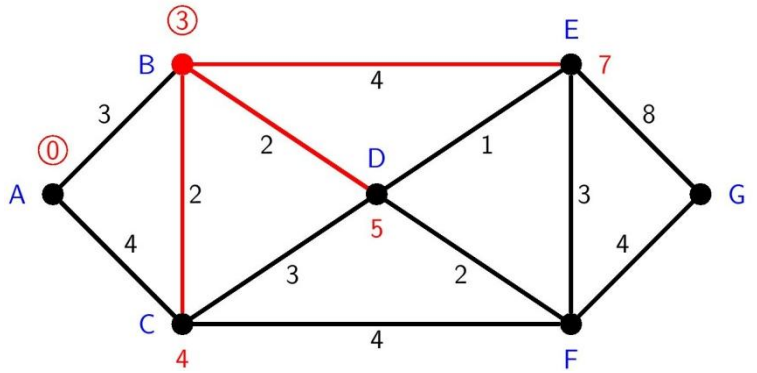
There are lots of little bits of information to keep track of when carrying out Dijkstra's algorithm. So that we don't get lost, possible distances will be written in red, and definite distances will be circled. All our vertices are unvisited. A is our origin vertex, so we mark its definite distance as 0. Then we calculate the possible distances to B and C from A:



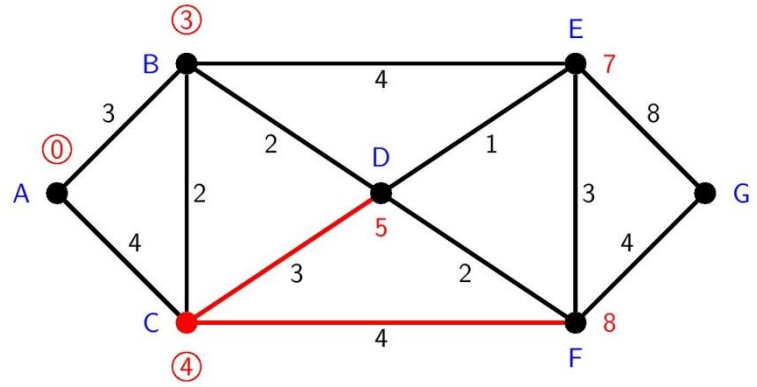
(Vertex A is in red because it is the current vertex. Edges (A,B) and (A,C) are in red because we are calculating distances along those edges.)
Now, step 4 tells us to find the smallest *possible distance* on the whole graph. We see that the smallest is 3 at vertex B. So B becomes the current vertex and its possible distance becomes definite:



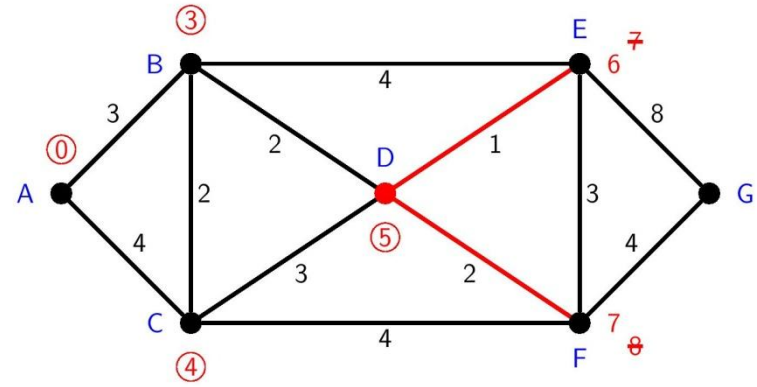
Now we calculate the distance from B to its unvisited neighbours. Note that we calculate a possible distance from B to C to be 3+2=5 which is larger than the possible distance at C already, so we don't replace it:



Now back to step 4: Find the smallest possible distance and make it definite. That's vertex C, which becomes the new current vertex. Then we calculate the possible distances to its unvisited neighbours:

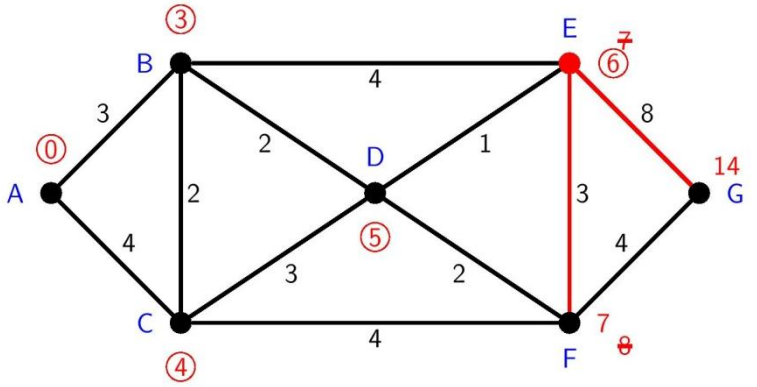


Note again that from C to D, the new possible distance was 4+3=7, so we don't replace the 5 that's already there. Now D has the smallest possible distance, so that becomes its definite distance and D becomes the current vertex:

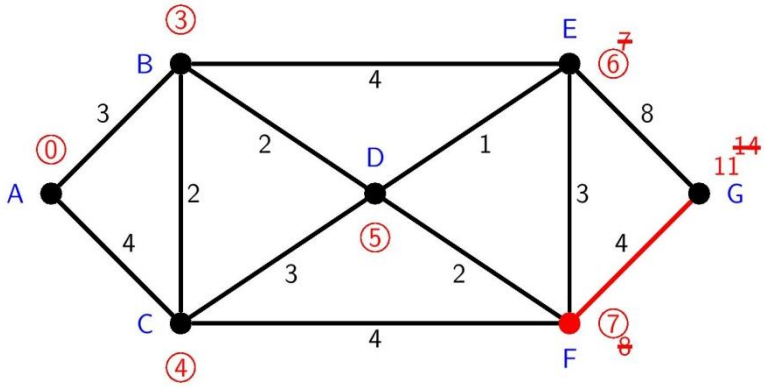


Note here that the new possible distances replace the old ones. So, we cross out the old ones and replace them with the new ones.

Now, what's the smallest? It's 6 at E, so that becomes the new current vertex.



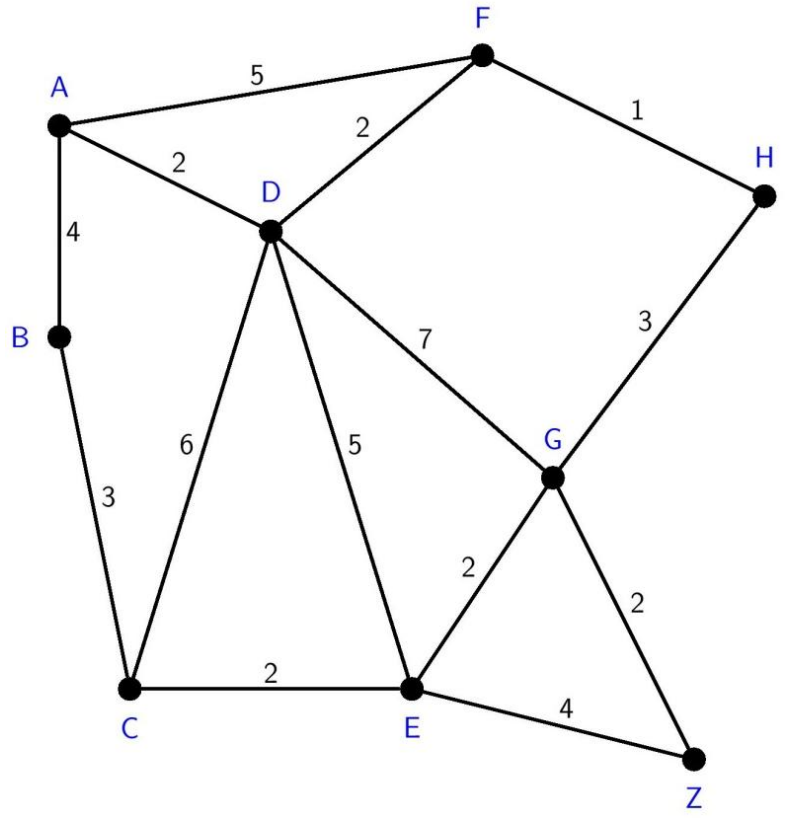
Now the smallest possible weight is 7 at F. This becomes the new current vertex:



Finally, G becomes the current vertex. We make its distance definite, and since G is the destination vertex, we STOP as it states in step 6.

So, what is the shortest distance from A to G? How can we work backwards to find the shortest *route* from A to G?

Let us practice Dijkstra's algorithm on the following graphs.

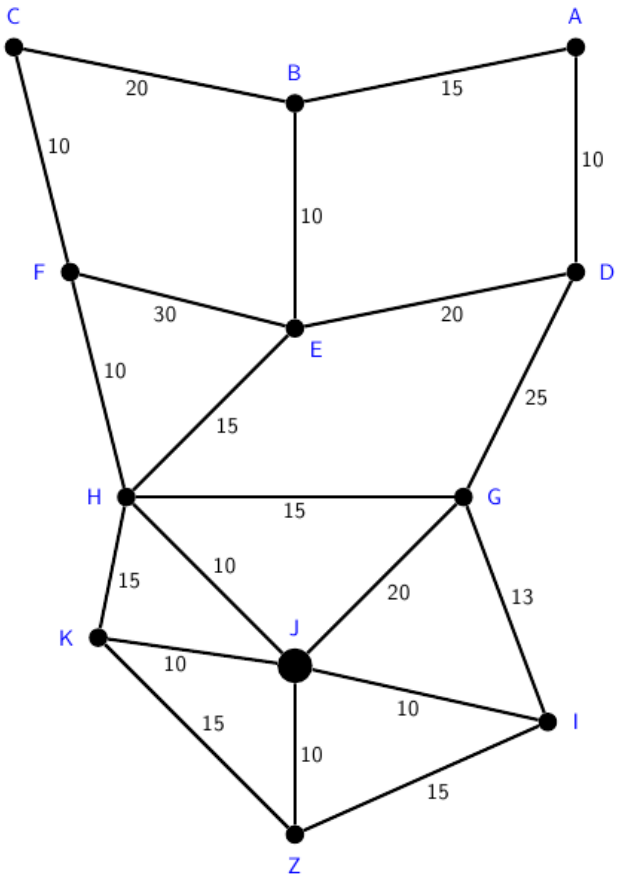


Assignment 4 – Dijkstra's Algorithm and Digraphs

Complete all homework on a separate sheet of paper.

Maths Questions

Use Dijkstra's Algorithm to find the shortest distance from A to Z on the following graph:



Research Question - Digraphs

Research what a *directed graph* is (sometimes called a *digraph*) and draw some examples. Then, answer the following problems:

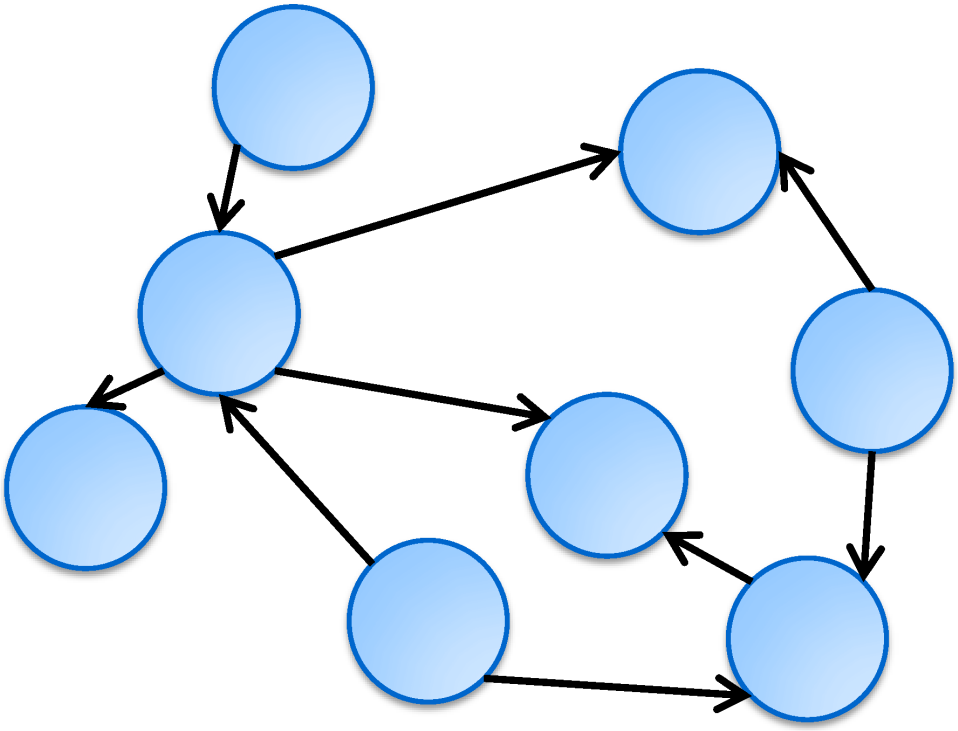
Provide some practical applications of digraphs. Why might we wish to use a digraph instead of a normal graph?

What would be different about walks, trails, and paths on a digraph?

What would be different about applying Dijkstra's algorithm on a digraph as opposed to a normal graph?

Aim for 250 words, include diagrams, and don't forget to give your sources!

Tutorial 5 – Advanced Dijkstra's Algorithm

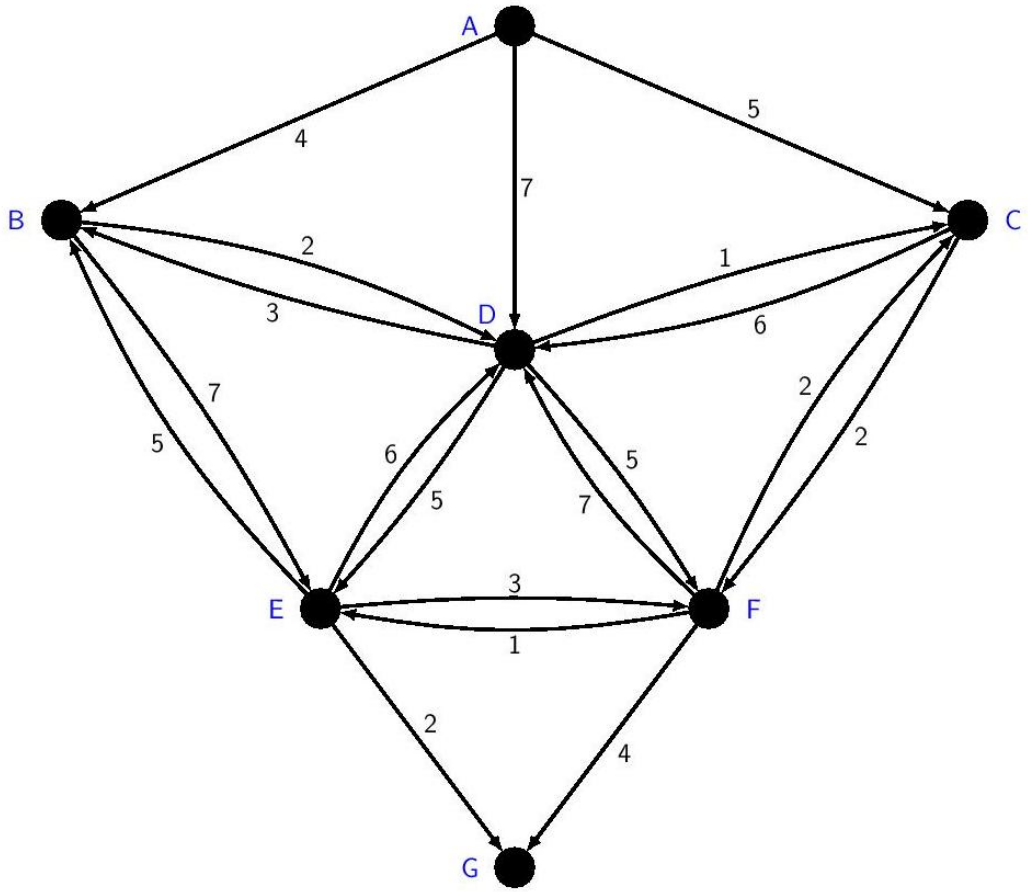


What is the Purpose of Tutorial 5?

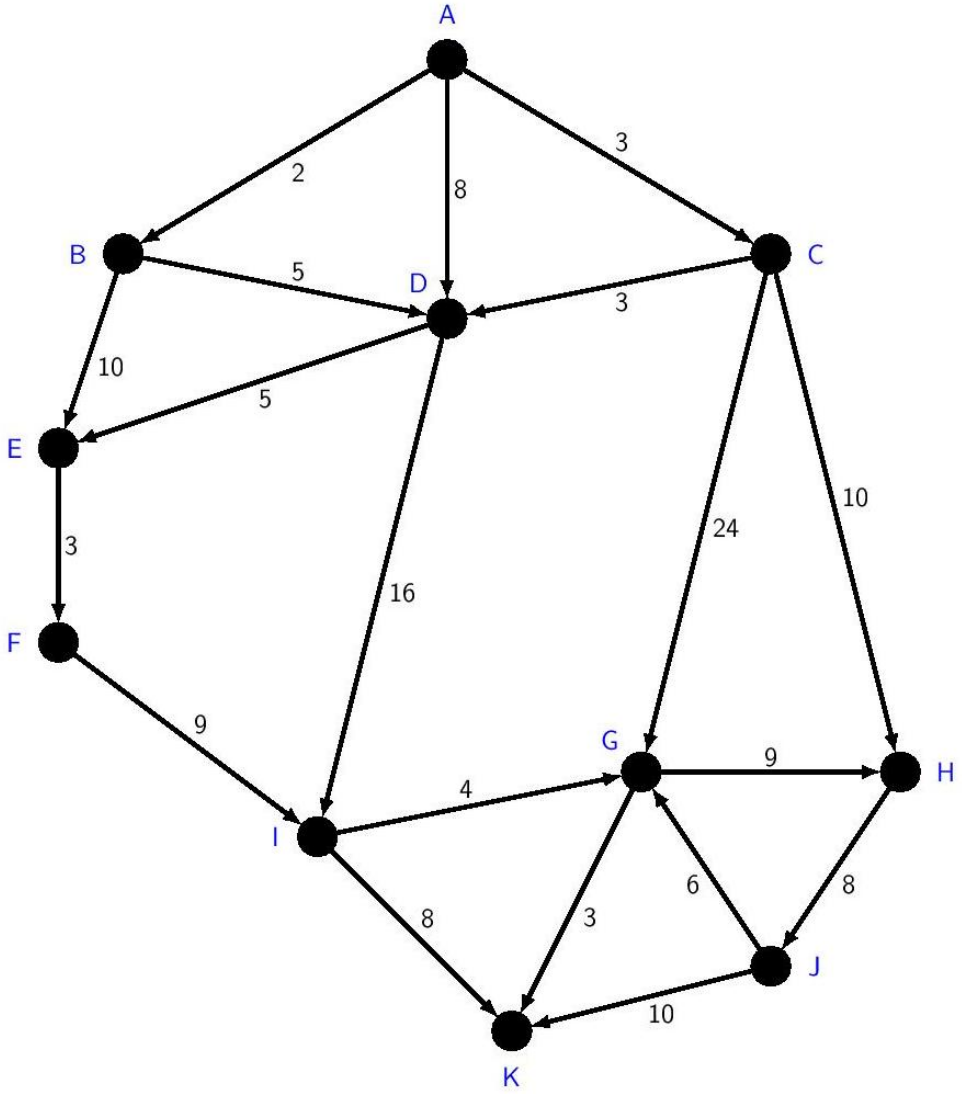
- To practice Dijkstra's algorithm on directed graphs
- To discuss the Final Assignment

Outline what you think will be the key differences between applying Dijkstra's algorithm on a directed graph as opposed to an undirected graph.

Let us practice Dijkstra's algorithm on the following two directed graphs.



This next example has fewer two-way edges. This resembles something like a one-way system of roads, often found in city centres.



We will now discuss the final assignment. Every pupil has the potential to write an outstanding final assignment – let us discuss some criteria to help you achieve this. Let us also take a moment to discuss the final assignment and any questions about it you might have.
The final assignment starts with a research question. Let us take a look at it now and discuss what an excellent solution will look like.

The rest of the final assignment is broken up into two parts: maths questions and a *final* question. Both of these sections are more mathematical than the research question, and may require you to demonstrate a different skill set in order for you to achieve an outstanding result. Let us discuss what an excellent solution to these problems will look like.

Final Assignment – Plan the Best Route

Research Question (30%) – DO THIS FIRST!

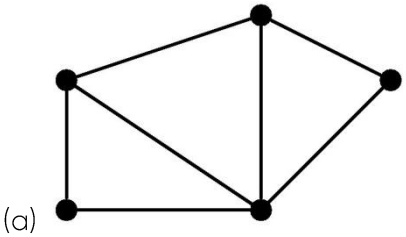
1. There is a famous problem in graph theory called the Chinese Postman Problem. Research this problem then describe it in your own words. Then give some applications of the problem (other than if you were a postman).
2. Now that you understand the Chinese Postman Problem, you must research how to solve it. The algorithm used it called Route Inspection. Research this algorithm and describe it in your own words.
3. Give a step-by-step set of instructions for how to apply the algorithm. This should look like the algorithms we've studied in tutorials, with inputs, outputs, and where to stop. Remember to refer back to Zeller's, Kruskal's, Dijkstra's algorithm to see the style in which algorithms are written.
4. Explain in your own words why the Route Inspection algorithm works and solves the Chinese Postman problem.

You should aim to write about 500 words for the research question. You are allowed to include diagrams.

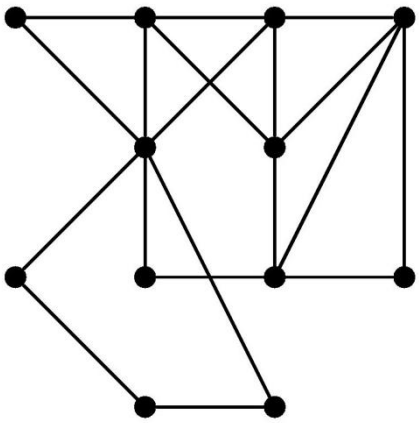
Maths Questions (45%)

To earn full marks on these questions, you must **show all your working** and **justify your answers**.

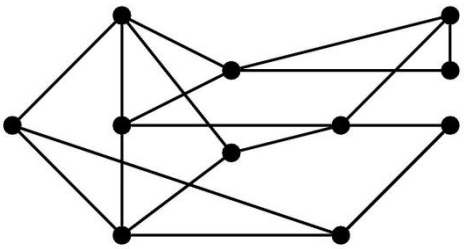
1. Which of the following graphs are Eulerian, semi-Eulerian, and non-Eulerian?



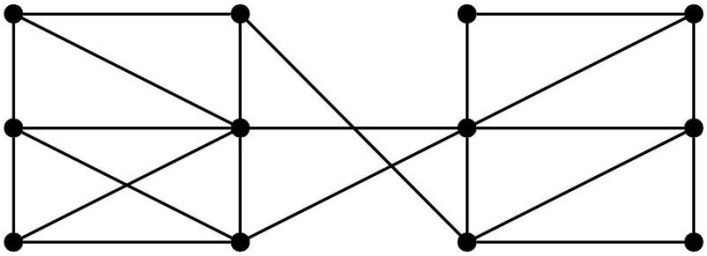
(a)



(b)



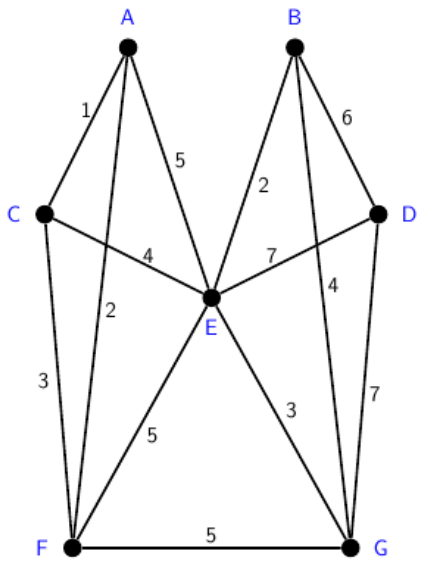
(c)



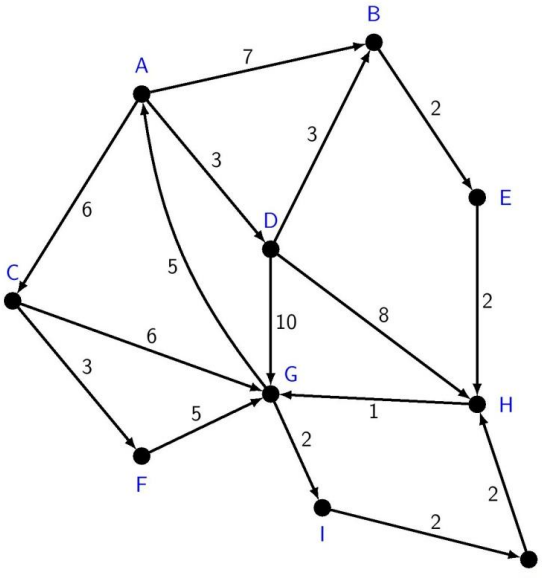
(d)

2. For the graphs in question 1 that are semi-Eulerian and non-Eulerian, what is the minimum number of edges you must add to the graph in order to make them Eulerian?

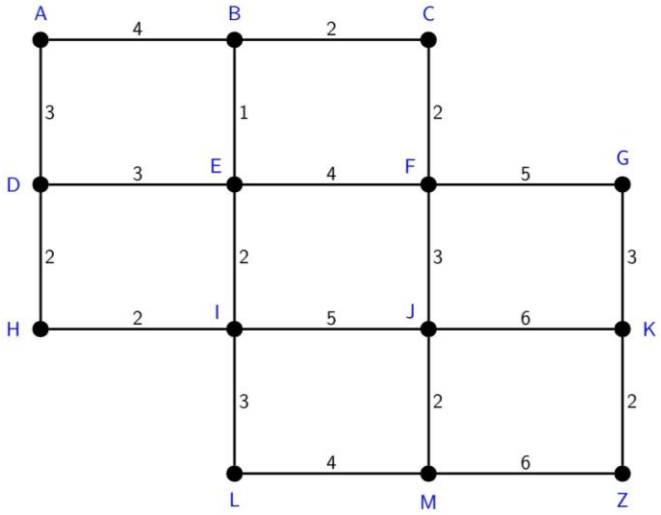
3. The graph below represents a school computer network. Each vertex represents a computer, and the edges represent wires which could be used to connect the computers. The weights represent the cost of installing the wires in tens of pounds – for example. The edge (A,E) has a weight of 5, so it will cost £50 to install this wire. What is the minimum cost of linking all the computers together?



4. Use Dijkstra's algorithm to find the shortest distance from A to Z on the following graphs.



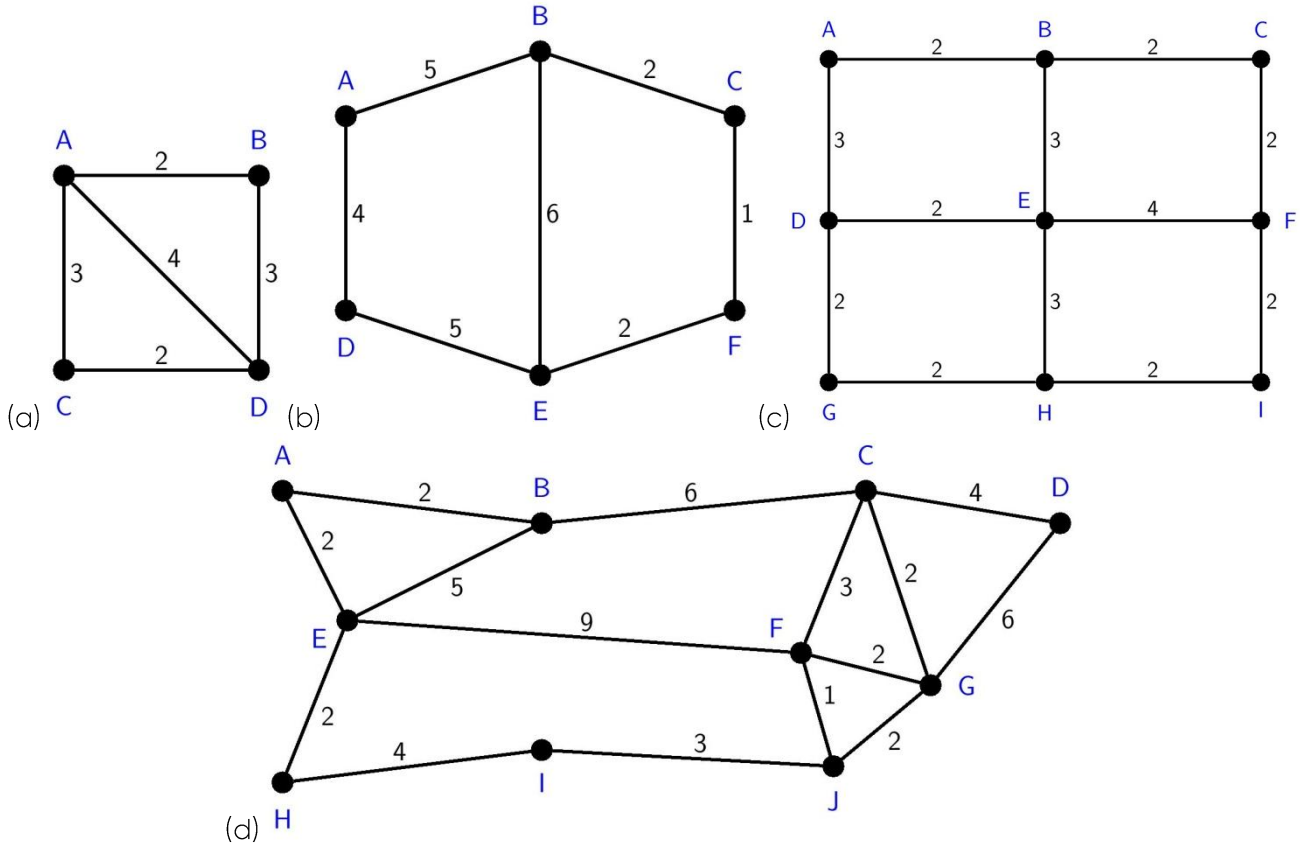
(a)



(b)

5. Is the shortest distance from A to Z always the same as the shortest distance from Z to A? Why/Why not?

6. Bill is a postman who must deliver letters to all houses on all the streets on four estates. Each estate is represented by one of the graphs at the top of the next page. Each street is an edge and the weights are the distances of each street. Use your knowledge of the Chinese Postman Problem to find the shortest distance Bill has to walk on each estate.



Final Question (25%) – A Day in the Life of a Courier

In this final question, you will take the role of a courier and plan out your daily activities. You have a very busy day ahead of you. The following steps must be done **as quickly as possible**.

- First, you must **travel from your hometown to a city** to deliver some parcels.
- You must then **walk around the depot** to shelve all the packages ready for their delivery the next day.
- Then you must **travel back to your hometown**, watching out for the rush hour traffic!
- Finally, calculate how long it took you (in minutes) **in total** to complete every task.

To earn all the marks, you must **show all of your working** and explain which method you are using at each step, and why that is the correct method.

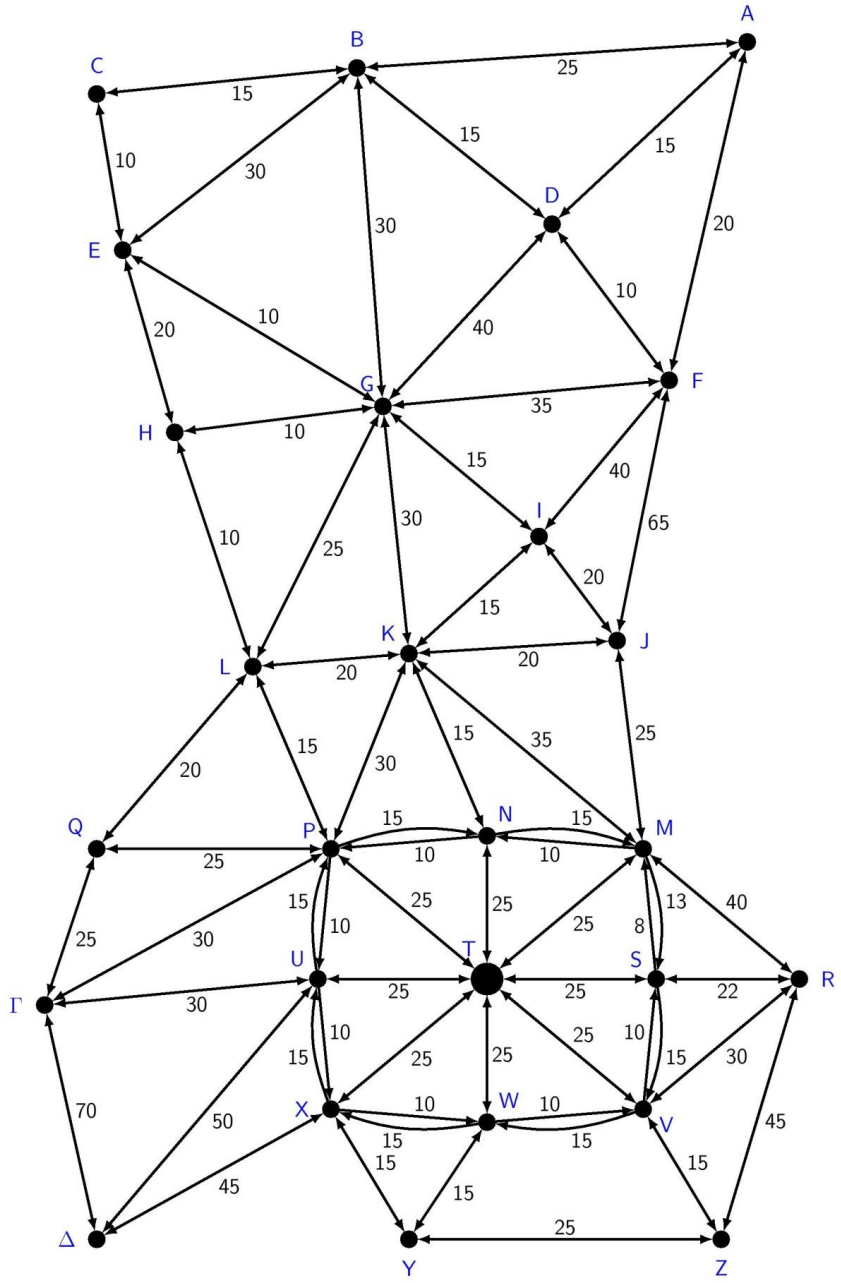
There are three graphs on the next few pages that you will need to solve this problem. Information about each of them is written below – read it carefully!

Graph 1 shows a digraph representing a map of the nation. Each vertex is a major city. Each edge is a road between two cities. The weights are the times (in minutes) it takes to drive along the roads. Note:

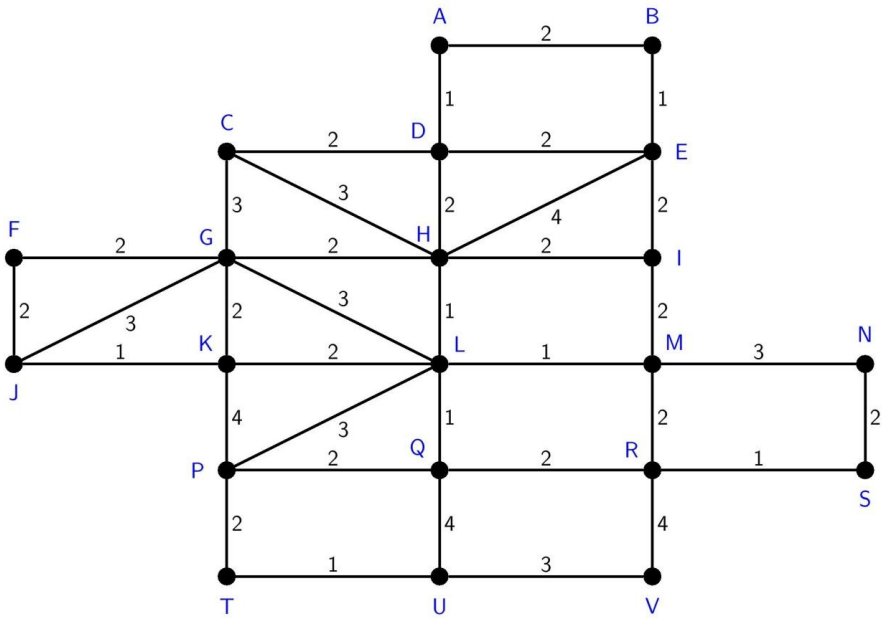
- This map has more vertices than letters in the alphabet, so two Greek letters, Γ (gamma) and Δ (delta) were used.
- Vertex **A** represents your **hometown** and **Z** represents your **destination city**.
- The path C-E-H-L-P represents a **motorway**. The cycle P-N-M-S-V-W-X-U-P represents a **ring road**. The vertex T represents the **capital city** of the nation.
- On the journey **from Z to A only**, there is more traffic. All edges connected the Capital city are increased in weight by 10. All edges in the ring road are increased in weight by 20. The motorway edges (P,L), (L,H), (H,E), and (E,C) are all increased in weight by 15.

Graph 2 represents a map of the depot in the destination city, Z. It's a huge warehouse filled with lots of carefully organised shelves, and all the packages must be stored correctly. Each edge represents a corridor in the depot. Each vertex represents a junction between the corridors. The weights are the times (in minutes) it takes to walk along the walkways. You have a lot of packages that need to be dropped off all over the depot, so you must **walk along every walkway at least once**.

Graph 1:



Graph 2:



Tutorial 6 – Feedback tutorial

What is the Purpose of Tutorial 6?

- To receive feedback on final assignments.
- To share examples of best practice with the other pupils in your group.
- To write targets for improvement in school lessons.
- To reflect on the programme including what was enjoyed and what was challenging.

Final assignment feedback

What I did well…	What I could have improved on…
•	•
•	•
•	•

My target for future work is…

Tutorial 7 – Feedback tutorial

What is the Purpose of Tutorial 6?

- To receive feedback on final assignments.
- To share examples of best practice with the other pupils in your group.
- To write targets for improvement in school lessons.
- To reflect on the programme including what was enjoyed and what was challenging.

Final assignment feedback

What I did well…	What I could have improved on…
•	•
•	•
•	•

My target for future work is…

Reflecting on Uni Pathways

What did you most enjoy about Uni Pathways?
• • •

What did you find challenging about the programme?	How did you overcome these challenges?
• • •	• • •

Appendix 1 – Referencing correctly

When you get to university, you will need to include references in the assignments that you write, so we would like you to start getting into the habit of referencing in your Brilliant Club assignment. This is really important, because it will help you to avoid plagiarism. Plagiarism is when you take someone else's work or ideas and pass them off as your own. Whether plagiarism is deliberate or accidental, the consequences can be severe. In order to avoid losing marks in your final assignment, or even failing, you must be careful to reference your sources correctly.

What is a reference?

A reference is just a note in your assignment which says if you have referred to or been influenced by another source such as book, website or article. For example, if you use the internet to research a particular subject, and you want to include a specific piece of information from this website, you will need to reference it.

Why should I reference?

Referencing is important in your work for the following reasons:

- It gives credit to the authors of any sources you have referred to or been influenced by.
- It supports the arguments you make in your assignments.
- It demonstrates the variety of sources you have used.
- It helps to prevent you losing marks, or failing, due to plagiarism.

When should I use a reference?

You should use a reference when you:

- Quote directly from another source.
- Summarise or rephrase another piece of work.
- Include a specific statistic or fact from a source.

How do I reference?

There are a number of different ways of referencing, and these often vary depending on what subject you are studying. The most important to thing is to be consistent. This means that you need to stick to the same system throughout your whole assignment. Here is a basic system of referencing that you can use, which consists of the following two parts:

1. A marker in your assignment: After you have used a reference in your assignment (you have read something and included it in your work as a quote, or re-written it your own words) you should mark this is in your text with a number, e.g. [1]. The next time you use a reference you should use the next number, e.g. [2].
2. Bibliography: This is just a list of the references you have used in your assignment. In the bibliography, you list your references by the numbers you have used, and include as much information as you have about the reference. The list below gives what should be included for different sources.

    a. Websites – Author (if possible), title of the web page, website address, [date you accessed it, in square brackets].

    E.g. Dan Snow, 'How did so many soldiers survive the trenches?', http://www.bbc.co.uk/guides/z3kgjxs#zg2dtfr [11 July 2014].

    b. Books – Author, date published, title of book (in italics), pages where the information came from.

    E.g. S. Dubner and S. Levitt, (2006) *Freakonomics*, 7-9.

    c. Articles – Author, 'title of the article' (with quotation marks), *where the article comes from* (newspaper, journal etc.), date of the article.

    E.g. Maev Kennedy, 'The lights to go out across the UK to mark First World War's centenary', Guardian, 10 July 2014.

Notes

Notes

Graph Theory: How Google Maps Plans Routes

Key Stage 4 Programme

researchersinschools.org