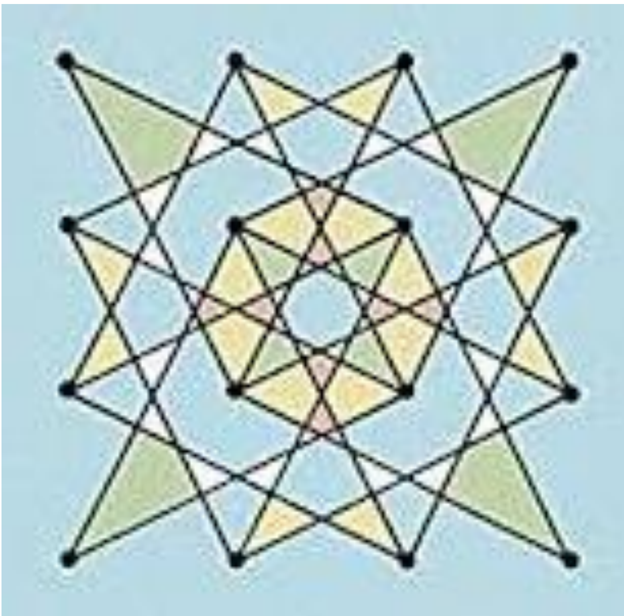


GRAPH THEORY RESEARCH CYCLE I

Group E2

ABSTRACT

Project Report for the 1st Research Cycle

CS 2052 Graph Theory for Computer Science

Group Members

Heshan Padmasiri

Bhanuka Dissanayake

Saranya Uthayakumar

Irosh Mayantha

Ranika Madurawe

The Problems

1) Show that any connected graph where the vertex degrees are even has an Eulerian cycle.
2) Show that any connected graph where there are exactly two vertices *a* and *b* of odd degree, there is an Eulerian path from *a* to *b*, but if there are more than two vertices of odd degree then it is impossible to construct an Eulerian path.

Definitions

- Degree: - The degree of a graph vertex is the number of graph edges which touch the said vertex (also known as *local degree* or *valency*) [1]
- Connected Graph: - A graph where there is a path from any point to any other point in the graph. [1]
- Eulerian cycle: - Is a trail which starts and ends at the same vertex. In other words it is a graph cycle which uses each graph edge exactly once. [1]
- Eulerian path: - A walk on the graph edges of a graph which uses each graph edge of the original graph exactly once. [1]

Proof

Problem (1)

We prove this using an algorithm which can find an Eulerian cycle for any even degree, connected graph, and proving its correctness.

THE ALOGRITHM

Part 1

Choose an arbitrary vertex $v \varepsilon V$ and draw a path from that vertex moving from vertex to vertex. As we travel through an edge we shall remove that edge from the graph until we return to the starting vertex *v*.

Part 2

When we have reached vertex *v* one of two outcomes could occur.

1) All edges have been traversed in which case we have found an Eulerian path
2) $\exists$ edges which have not yet been traversed in the path found in part 1

Assume (2) has occurred.

This would imply that whatever edges remaining consist of vertices (which have a degree larger than 0) have an even degree.

So, we repeat part 1 using a vertex which still has edges connected to it. And repeat part 2 until we have removed all the edges in the graph.

Part 3

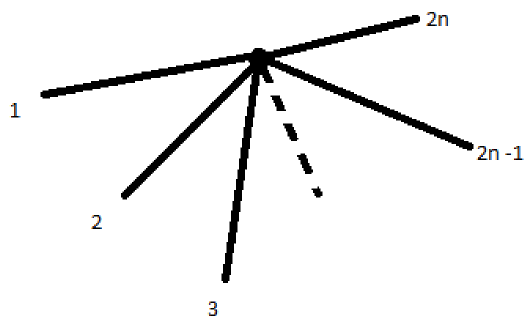
Now we have a set of set of edge disjoint cycles with at least 2 of them having a common vertex.

Now we combine these two cycles by inverting the cycle starting and ending at the common vertex in the 2 chosen cycles and insert it into the other generating one larger cycle. We reduce the set of cycles until we have one large cycle consisting of all edges. So, we have an Eulerian cycle.

PROOF OF CORRECTNESS

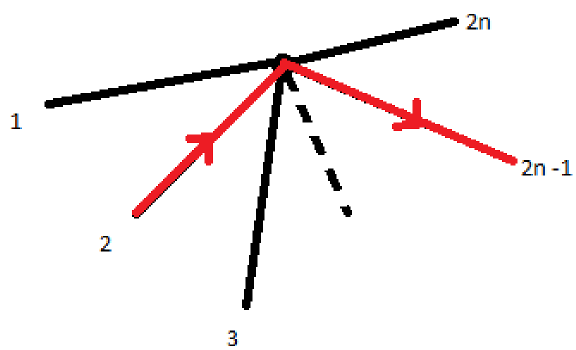
<u>Part 1</u>

As if we have visited said vertices we would have arrived and left using 2 unique edges removing them from the graph and reducing the degree of the said vertices by 2 each time we visit it.



Here $n \varepsilon \mathbb{R}$.

Each time we visit a vertex and leave the vertex.



Degree of node $v = 2n - 2$.

$= 2(n - 1) \varepsilon \mathbb{R}$

Degree is still even

If we have yet to visit the vertex the degree of the vertex is still even.

By visiting or revisiting a graph the vertex degree will reduce from

$$2n, 2n - 2, 2n - 4, \ldots\ldots\ldots, 4, 2, 0.$$

And when the degree becomes 0 that vertex cannot be visited once more.

Unless we have returned to the starting index we cannot get stuck as when we start drawing the path we have only exited from the starting node only reducing its degree value by one.

∴ This path cannot get stuck at any vertex other than the starting vertex

∴ ∃ a path which starts and ends on the starting vertex

<u>Part 2</u>

No need for proof as all this part involves an iteration of part 1

<u>Part 3</u>

We shall prove the correctness of part 3 using contradiction.

Let's say that there aren't at least 2 cycles which share a common vertex. -> (1)

We have used all the edges exactly once to create the cycles in the set of cycles finally obtained using parts 1 and 2. There cannot exist a path which connects the cycles the final set of cycles obtained.

However, statement (1) says that there are no common vertices which connects the two paths either.

⇒ The graph is not connected ↯

Our assumption (1) is wrong.

∴ There exists at least 2 cycles which share a common vertex

This proves that our algorithm is correct and will always find an Eulerian cycle provided that the Graph is connected and every node has an even degree

<u>Problem (2)</u>

PART A

We will once again provide an algorithm to find the Eulerian path and prove its correctness.

Let us assume the two nodes with odd degrees are *a* and *b*. (Where $a, b \, \varepsilon \, \mathbb{V}$)

THE ALGORITHM

<u>Part 1</u>

Choose one of the odd vertices. Let us assume the chosen vertex is *a*. We will draw a path from *a* moving from vertex to vertex until we reach vertex *b*. As we travel through an edge we shall remove that edge from the graph until we return to the *b*.

<u>Part 2</u>

When we have reached vertex *b* one of two outcomes could occur.

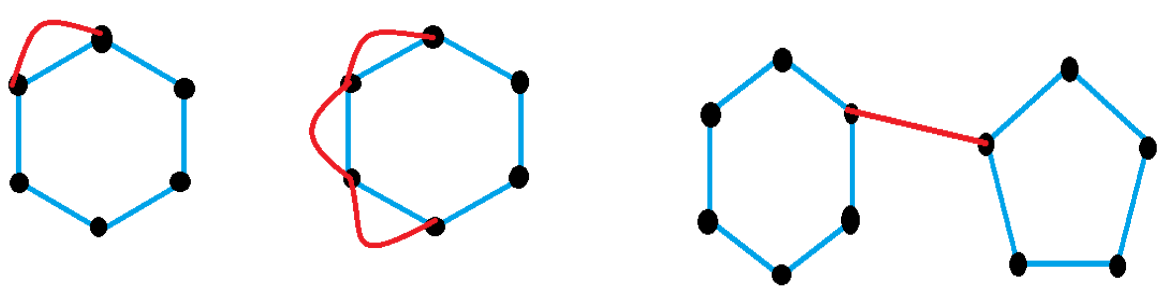
1) All edges have been traversed in which case we have found an Eulerian path connecting *a* and *b*.
2) ∃ edges which have not yet been traversed in the path found in part 1

Assume (2) has occurred.

It will be seen that the remaining edges will create a Eulerian cycle or Eulerian cycles if not connected.

So, combining the path and the cycles at their common points will result in an Eulerian path which connects vertices *a* and *b*.

<u>Examples</u>



PROOF OF CORRECTNESS

Part 1

As stated in Problem (1) Part (1) we cannot get stuck at a node with an even degree. However, in this case there exists 2 vertices with odd degrees, Namely *a* and *b*.

Let us assume *a* has a degree of $2n + 1$ where $n \varepsilon \mathbb{R}$. Since we start and leave vertex *a* that will reduce its degree by 1. Each time we revisit vertex *a* it's degree will reduce by 2. So it's degree will reduce in the following pattern

$$2n+1,\ 2n,\ 2n-2,\ 2n-4,\ \ldots\ldots\ldots,\ 4,\ 2,\ 0.$$

When the degree of *a* becomes 0, it cannot be visited once more. So, there is no way we can get stuck at *a*.

We can though get stuck at *b*. However, as we are calculating a path connecting *a* and *b* getting stuck at vertex b is irrelevant.

This implies we cannot get stuck at a vertex with an even degree neither can we get stuck at vertex *a*. So, we can definitely find a path which connects a and b

Part 2

Let's assume that we have not found the Eulerian path in part 1. i.e. we have not traversed through all edges in the graph.

a and *b* have odd degrees and we are found a path which starts at *a* and ends at *b*.

The path connecting the two will reduce the degree of a and b by $2x_i - 1$. Where x_i is the number of times a vertex i is revisited ($i = a \text{ or } b$)

This is an odd number for any x_i. So, the degree of *a* and *b* in the remaining graph which is

$$2n - 1 - (2x_i - 1) = 2\,(n - x_i)$$

will be even. The other vertices will always have an even degree as proved in Problem (1) Part (1)

∴ The remaining edges will form a graph consisting only of vertices which have an even degree.

⇒ According to the first problem there exists a Eulerian path in this remaining graph.

However according to Problem (1) Part (3), There exists a common vertex for the path and Eulerian cycle obtained. By inserting them into one another we receive a Eulerian path connecting *a* and b.

⇒ In a graph which has 2 vertexes with odd degrees we can draw an Eulerian path connecting the two vertices with odd degrees

PART B

<u>Part 1</u>

Let's consider a graph which has more than 2 vertices with odd degrees. Similar to part A in this problem we shall separate a path which connect any random 2 such vertices. We shall name them *s* and *t*.

<u>Part 2</u>

It can be seen that in the remaining portion of the graph there will exist at least 1 subgraph which has more or exactly 2 vertices with odd degrees.

i) If this subgraph has exactly 2 vertices with odd degrees according to part A we have proven that there exists a Eulerian path which connects the 2 vertices with odd degrees. Therefore, the complete graph (including the part removed in part 1) cannot have an Eulerian path connecting *s* and *t*.

ii) Let's assume that the subgraph has more than 2 vertices with odd degrees. Then according to part 1 we will once again divide this subgraph to smaller subgraphs. Until we find a subgraph which has exactly 2 vertices with odd degrees using the proof in (i) and working upwards we can prove that the complete graph has no Eulerian path connecting the vertices with odd degrees.

Question 1

Siripala, who has just learnt the importance of a strong password has decided to use one for his Facebook account. However, since he can’t remember the password correctly he has decided to write it down.

To prevent somebody who might steal his notes and access his Facebook profile he has decided to write down every 3-character long substring of his password separately and mixes them up. So, by looking at them he might remember the password but not the thief. He has written *n* such substrings. repeated substrings represent repeated substrings in the password which is *n*+2 characters long

However, when he tried to login to Facebook again he couldn’t remember his password. You as his best friend have volunteered to help Siripala break his own cipher.

*Important

1) Siripala might have written down his password incorrectly and there may be no password satisfying all of his substrings

2) There may be more than 1 password that matches the given substrings. In that case you may find any valid password

Test case 1

aca
aba
aba
cab
bac

OUTPUT : - Yes abacaba

Test Case 2

cb1
bCb
cb1
b13

OUTPUT : - No

Test Case 3

aaa
aaa
aaa
aaa
aaa
aaa
aaa

OUTPUT :- Yes aaaaaaaaa

Question 2

In the strange land of JAVASE there is an even stranger park called Whatthefunction. It has only one gate common for entry and exit. You have to use an 'Hyper advanced step climbing robot' to travel in this park.

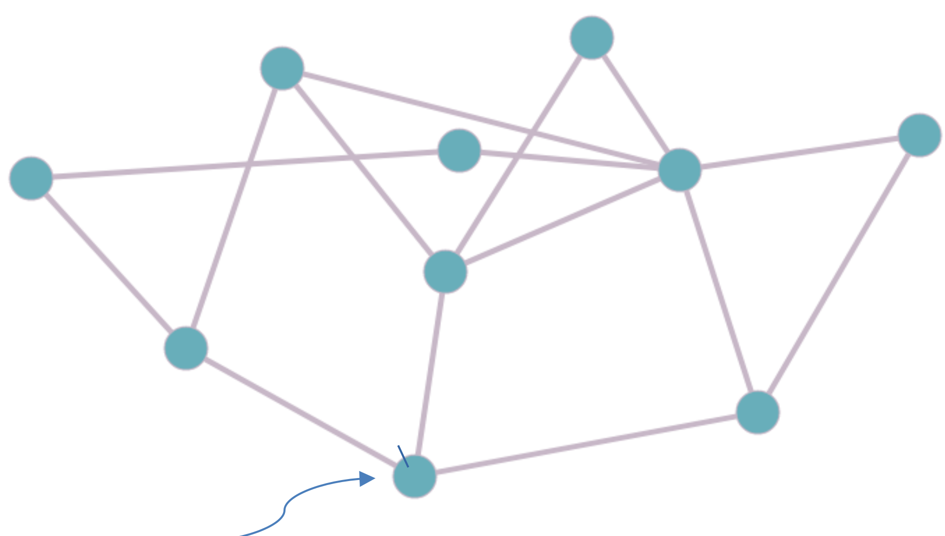
Malkumaraya and Malkumari have chosen this park for their honeymoon. Malkumari has manage to persuade Malkumaraya to visit Whatthefunction park, hoping to spend as much time as possible in the park. Malkumari promises that they will visit one path just once. (You forge as many paths between junctions using the robot) and that they will at least travel on all the paved roads.

However, Malkumaraya has more important things to do during the night than looking at flowers. But he also doesn't want to say no to his wife. So, he has to set out to find a way to meet all the requirements of his wife while spending as little time as possible in the park. At the entrance of the park Malkumaraya sees the Map of the park. Can you help him find a path where you use all the pave roads and spend as little time possible in the Park?

Note that

*If you use the paved road between two junctions the robot can make it in 5 minutes

*If you forge your own path (travel outside of the paved roads) you can get to your destination in 10 min



Entrance to the park is

Question 3

Each police officer in the city of Portland is given a section of the city to patrol during the night. The officer must visit every road of his patrol area before his shift ends

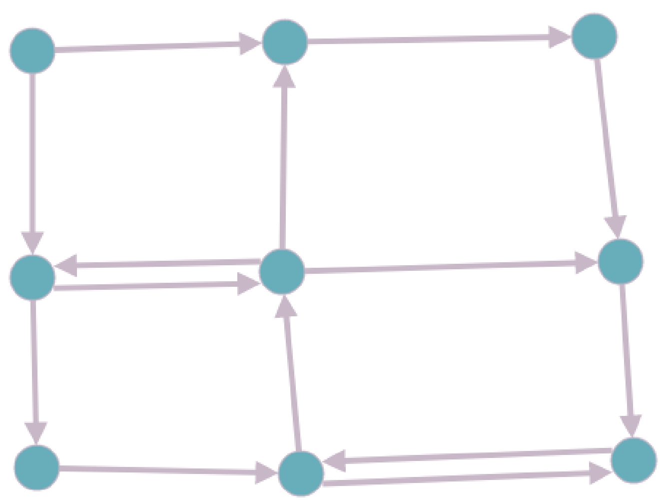
However, to reduce traffic jams in the city, the city has come up with a traffic scheme where each road has been made into a one-way road. You a police officer, One day arrive to your shift late and you realize that there will not be enough time to conduct your patrol by visiting the same road multiple times

You must figure out a way to conduct your patrol without breaking the law and travel through each road only once.

Given below is the map of your patrol area.

Note that

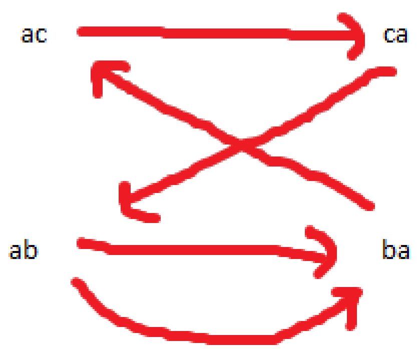
*You can travel along any stretch of road using your vehicle and you cannot walk



<u>**Solution -1-**</u>

Let substring be S, represent it as two vertices s[0]s[1] and s[1]s[2] and added an directed edge from s[1] to s[1] find an Eulerian path in the graph

<u>Test Case 1</u>

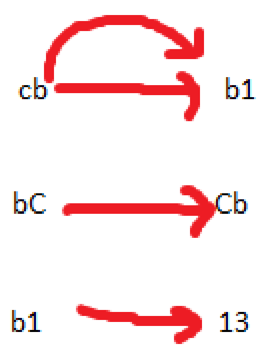


In this test case there exists a Eulerian path (As there are 2 vertices with odd edges There exists a path between these two vertices)

ab -> ba -> ac -> ca -> ab -> ba

OUTPUT: - abacaba

<u>Test Case 2</u>



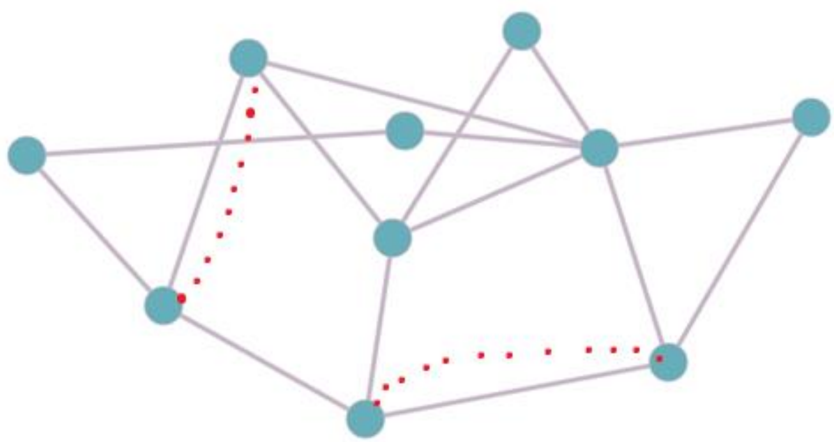
The graph is not connected so no Eulerian path exists. Therefore no valid password can be derived

This can be similarly shown for any other test case.

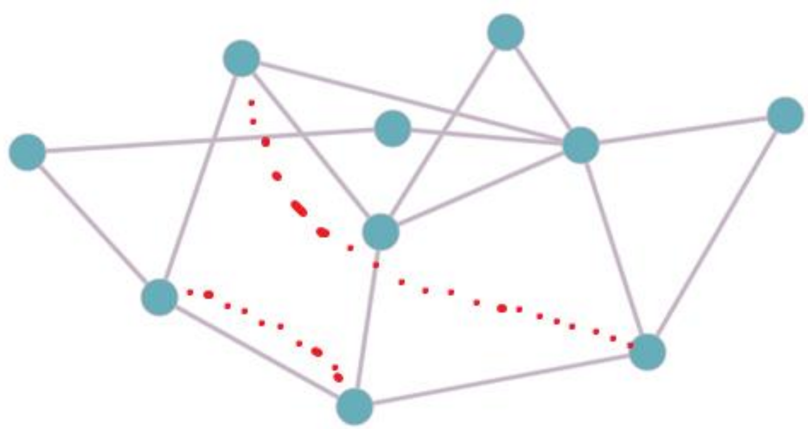
Solution -2-

To leave the park as quickly as possible while meeting Malkumari's requirements we must find an Eulerian cycle. It would be in the interest of Malkumara to forge as little as possible paths using the robots as that would take the most time.

In this example there exists 4 junctions with odd degrees. If we use the robot to pair up these 4 vertices as shown below we shall get a graph which only has vertices with even degrees.



This solution is also acceptable.



In a general example we will notice that if there exist junctions with odd degrees that they will always exist in multiples of 2. In which case we will forge paths with the robot to make these junctions have even degrees

Solution -3-

We have to find whether there exists a Eulerian cycle or not.

References

[1]https://wolfram.com/mathematica