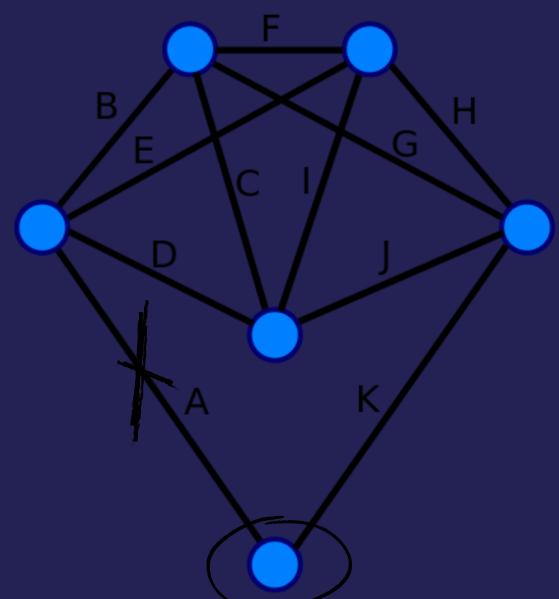
Algoritmo de Fleury

Joel Victor, Raynan Serafim



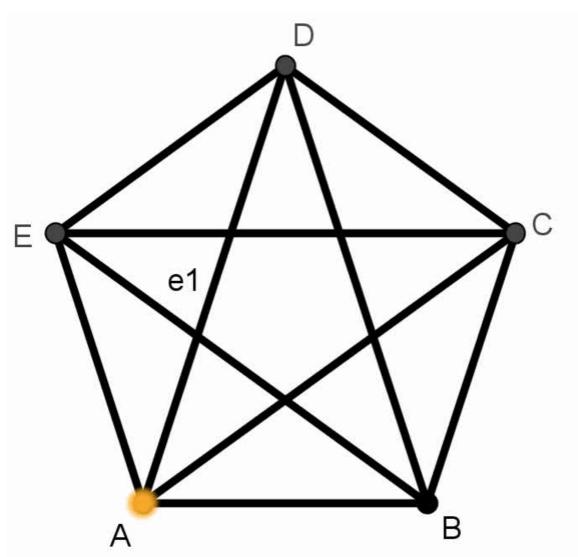
Trilha e Ciclo(Circuito) Euleriano(a)



- Presentes em grafos não dirigidos e conexos
- Trilha é o caminho em que todos os vértices do grafo são visitadas
- Uma trilha que possui um ciclo é chamada de ciclo Euleriano



Trilha e Ciclo(Circuito) Euleriano(a)

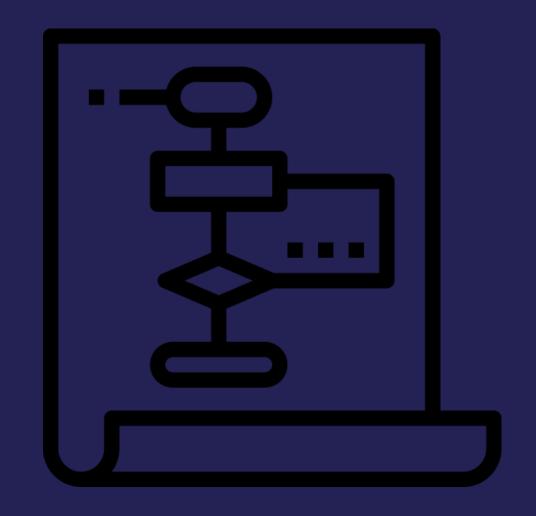


- Para a formação de uma trilha, é preciso que o número de vértices com grau impar, seja 0 ou 2
- Caso o grafo possua 2 vértices de grau impar, o inicia em uma delas e termina na outra
- Caso o grafo n\u00e3o possua v\u00e9rtices impares, \u00e9 poss\u00edvel criar um ciclo Euleriano a partir de qualquer v\u00e9rtice



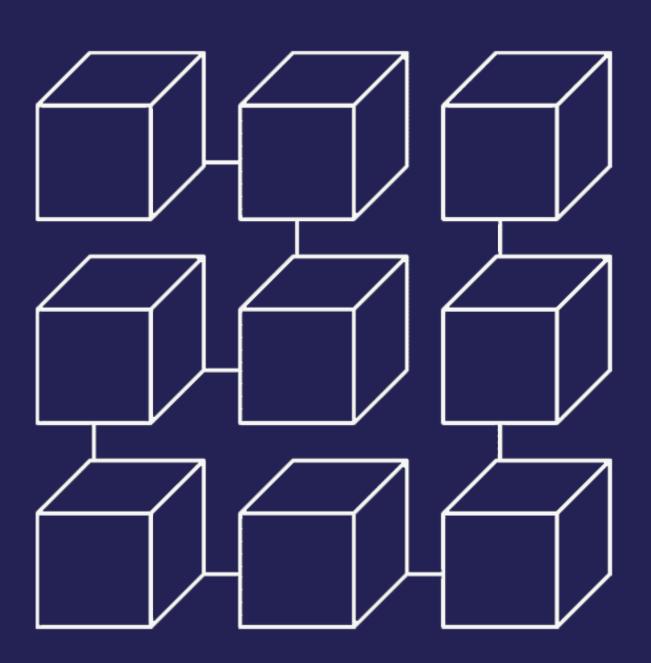
Algoritmo de Fleury

- É um algoritmo elegante e simples que data de 1883.
- É usado para descobrir se um grafo possui as características de um grafo Euleriano e produzir trilhas ou ciclos possíveis.





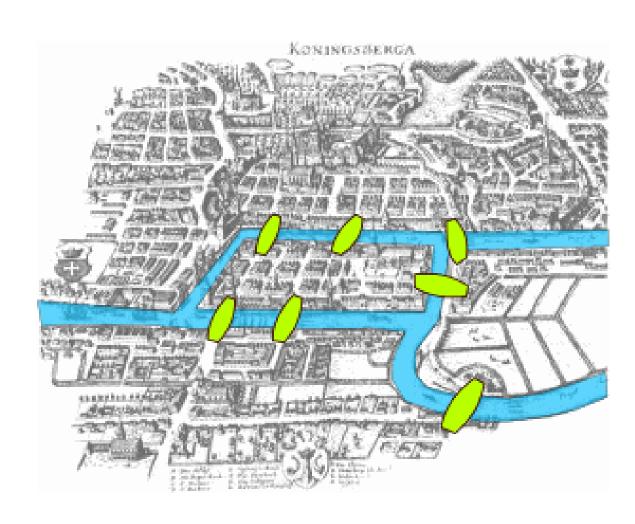
Estrutura de dados



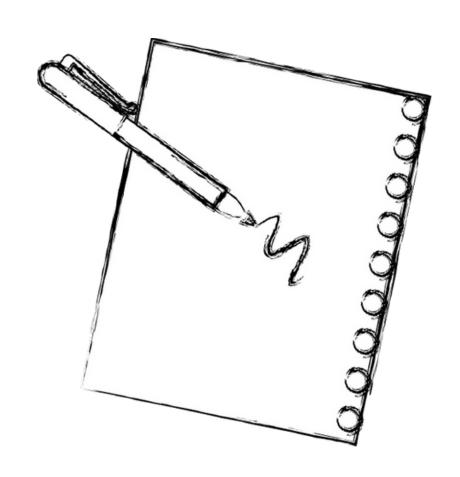
- Matriz de adjacência
- Estrutura de pilha para de algoritmo de DFS
- Variaveis diversas

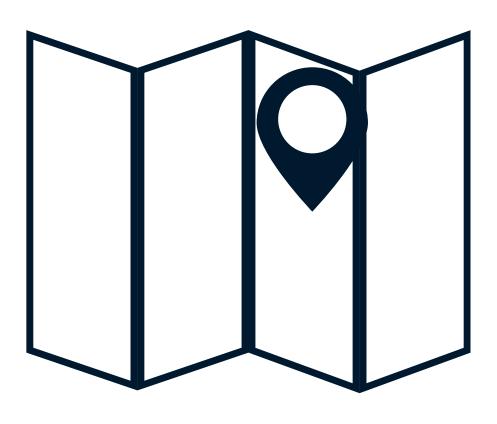


Aplicações

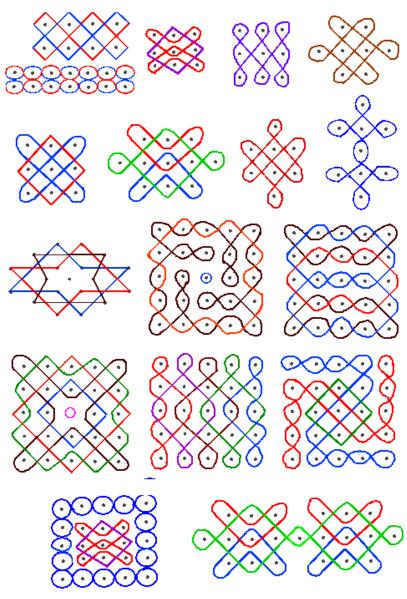


- Analise de problemas classicos
 - Problema das 7 pontes de Königsberg
 - Traçar diagrama de pontos em uma folha
 - o Problemas de inspeção de rota



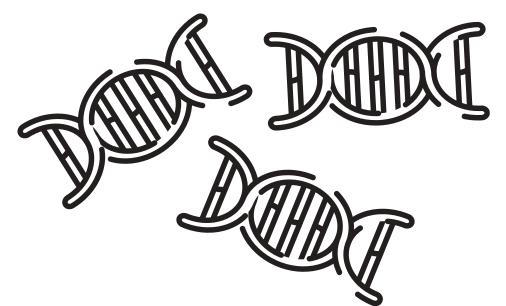


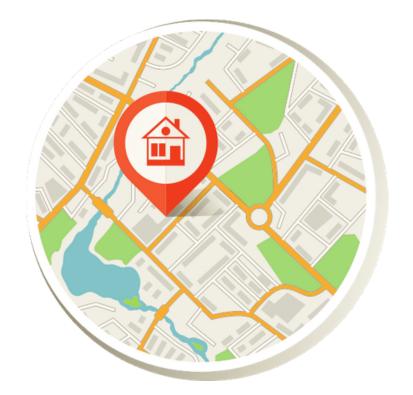




Aplicações

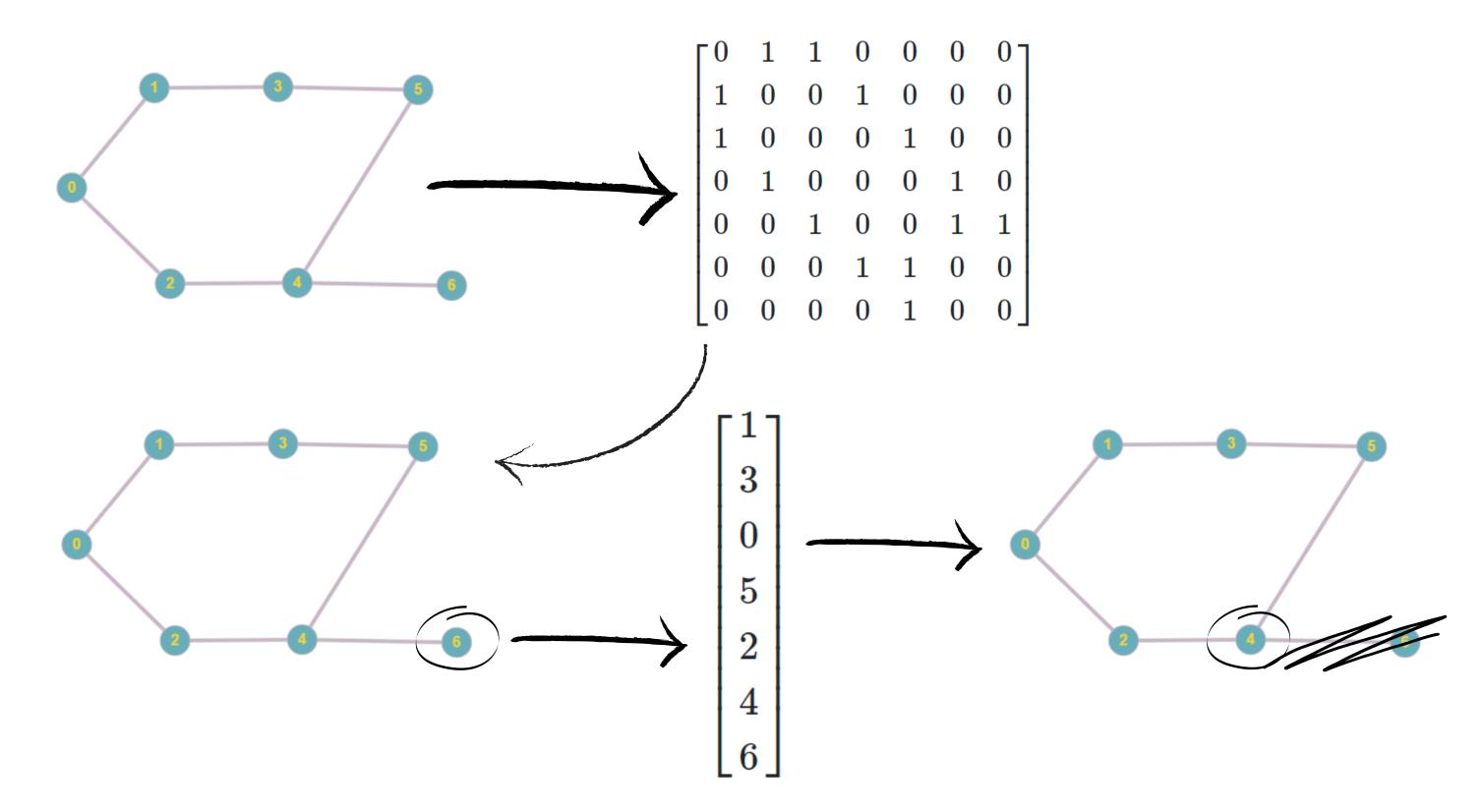
- Anexando informações a cada vertice, é possivel resolver alguns tipos de problemas que podem ser representados com grafos eulerianos
 - Reconstrução de fragmentos de DNA na área de Biologia Computacional
 - Geração de rotas para entregas em umo representação em grafos de uma cidade
 - Geração de padrões de desenho Kolam



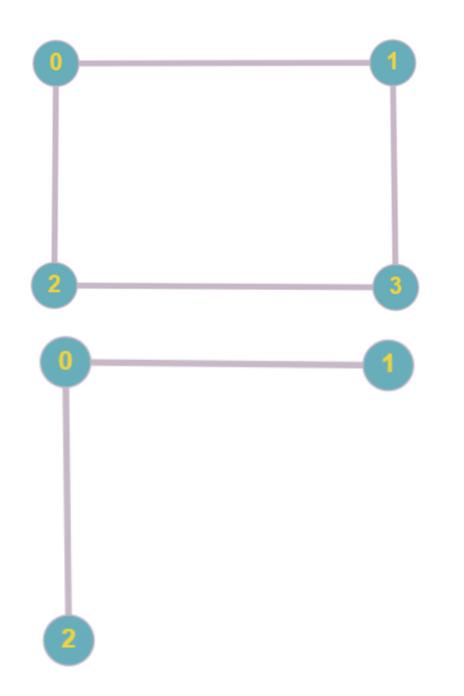




Algoritmo





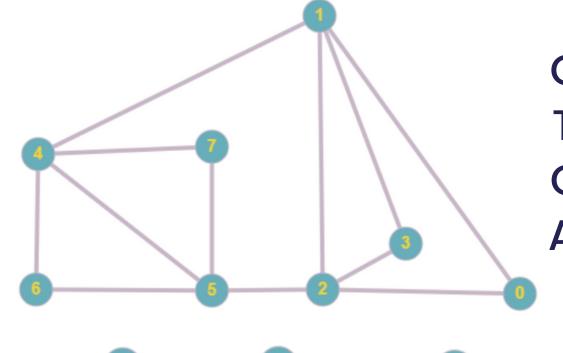


Graph Online: $0 \rightarrow 1 \rightarrow 3 \rightarrow 2 \rightarrow 0$ Tutorials Point: $0 \rightarrow 1 \rightarrow 3 \rightarrow 2 \rightarrow 0$ Geeks for Geeks: $0 \rightarrow 2 \rightarrow 3 \rightarrow 1 \rightarrow 0$

Autoria: $0 \rightarrow 1 \rightarrow 3 \rightarrow 2 \rightarrow 0$

Graph Online: $2 \rightarrow 0 \rightarrow 1$ Tutorials Point: $1 \rightarrow 0 \rightarrow 2$ Geeks for Geeks: $1 \rightarrow 0 \rightarrow 2$ Autoria: $1 \rightarrow 0 \rightarrow 2$



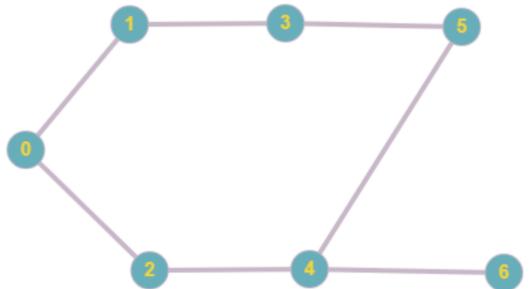


G.O: 0->1->3->2->5->6->4->7->5->4->1->2->0

T.P: 0->1->2->3->1->4->5->6->4->7->5->2->0

G.G: 0->1->3->2->1->4->6->5->4->7->5->2->0

Autoria: 0->1->2->3->1->4->5->6->4->7->5->2->0

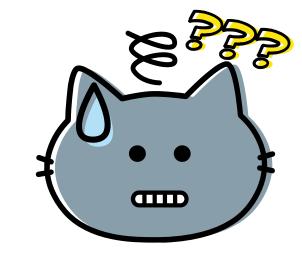


Graph Online: 6 -> 4 -> 2 -> 0 -> 1 -> 3 -> 5 -> 4

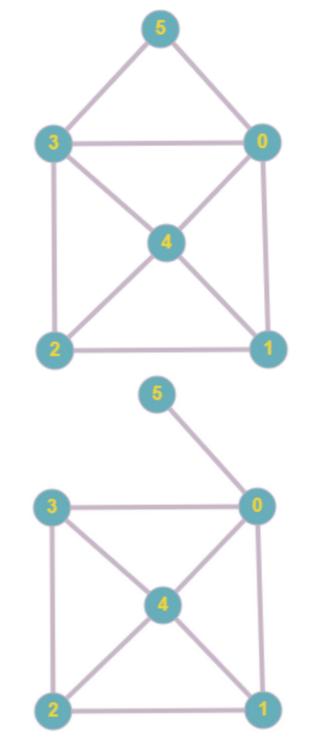
Tutorials Point: 4 -> 2 -> 0 -> 1 -> 3 -> 5 -> 4 -> 6

Geeks for Geeks: 4 -> 2 -> 0 -> 1 -> 3 -> 5 -> 4 -> 6

Autoria: $4 \rightarrow 2 \rightarrow 0 \rightarrow 1 \rightarrow 3 \rightarrow 5 \rightarrow 4 \rightarrow 6$







G.O:
$$2->1->0->3->2->4->0->5->3->4->1$$

T.P:
$$1->0->3->2->1->4->0->5->3->4->2$$

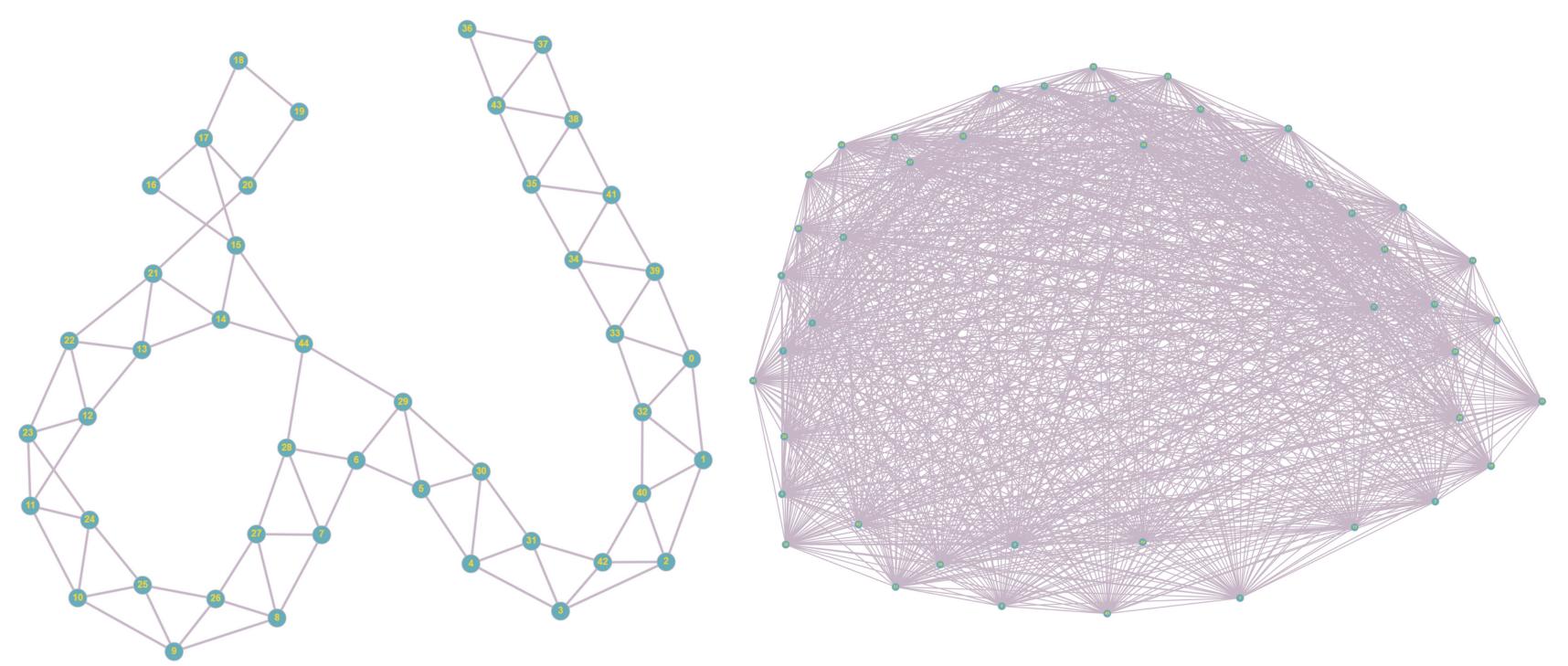
G.G:
$$1->0->3->2->1->4->3->5->0->4->2$$

Graph Online: Graph has not Eulerian path

Geeks for Geeks: Loop infinito!

Autoria: O grafo não possui trilha euleriana!







0 -> 44 -> 43 -> 42 -> 44 -> 41 -> 43 -> 40 -> 44 -> 39 -> 43 -> 38 -> 44 -> 37 -> 43 -> 36 -> 44 -> 35 -> 43 -> 34 -> 44 -> 33 -> 43 -> 32 -> 44 -> 31 -> 43 -> 30 -> 44 -> 29 -> 43 -> 28 -> 44 -> 27 -> 43 -> 26 -> 44 -> 25 -> 43 -> 24 -> 44 -> 23 -> 43 -> 22 -> 44 -> 21 -> 43 -> 20 -> 44 -> 19 -> 43 -> 18 -> 44 -> 17 -> 43 -> 16 -> 44 -> 15 -> 43 -> 14 -> 44 -> 13 -> 43 -> 12 -> 44 -> 11 -> 43 -> 10 -> 44 -> 9 -> 43 -> 8 -> 44 -> 7 -> 43 -> 6 -> 44 -> 5 -> 43 -> 44 -> 3 -> 43 -> 2 -> 44 -> 1 -> 43 -> 0 -> 42 -> 41 -> 40 -> 42 -> 39 -> 41 -> 38 -> 42 -> 37 -> 41 -> 36 -> 42 -> 35 -> 41 -> 34 -> 42 -> 33 -> 41 -> 32 -> 42 -> 31 -> 41 -> 30 -> 42 -> 29 -> 41 -> 28 -> 42 -> 27 -> 41 -> 26 -> 42 -> 25 -> 41 -> 24 -> 42 -> 23 -> 41 -> 22 -> 42 -> 21 -> 41 -> 20 -> 42 -> 19 -> 41 -> 18 -> 42 -> 17 -> 41 -> 16 -> 42 -> 15 -> 41 -> 14 -> 14 -> 12 -> 42 -> 11 -> 41 -> 10 -> 42 -> 9 -> 41 -> 8 -> 42 -> 7 -> 41 -> 41 -> 41 -> 42 -> 6 -> 42 -> 5 -> 41 -> 4 -> 42 -> 3 -> 41 -> 2 -> 42 -> 1 -> 41 -> 0 -> 40 -> 39 -> 38 -> 40 -> 37 -> 39 -> 36 -> 40 -> 35 -> 39 -> 34 -> 40 -> 33 -> 39 -> 32 -> 40 -> 31 -> 39 -> 30 -> 40 -> 29 -> 39 -> 28 -> 40 -> 27 -> 39 -> 26 -> 40 -> 25 -> 39 -> 24 -> 40 -> 23 -> 39 -> 22 -> 40 -> 21 -> 39 -> 20 -> 40 -> 19 -> 39 -> 18 -> 40 -> 17 -> 39 -> 16 -> 40 -> 15 -> 39 -> 15 -> 39 -> 16 -> 40 -> 15 -> 39 -> 16 -> 40 -> 15 -> 39 -> 16 -> 40 -> 15 -> 39 -> 16 -> 40 -> 17 -> 39 -> 16 -> 40 -> 17 -> 39 -> 16 -> 40 -> 17 -> 39 -> 16 -> 40 -> 17 -> 39 -> 16 -> 40 -> 17 -> 39 -> 16 -> 40 -> 17 -> 39 -> 16 -> 40 -> 17 -> 39 -> 16 -> 40 -> 17 -> 39 -> 16 -> 40 -> 17 -> 39 -> 16 -> 40 -> 17 -> 39 -> 16 -> 40 -> 17 -> 39 -> 16 -> 40 -> 17 -> 39 -> 16 -> 40 -> 17 -> 39 -> 16 -> 40 -> 17 -> 39 -> 16 -> 40 -> 17 -> 39 -> 16 -> 40 -> 17 -> 39 -> 16 -> 40 -> 17 -> 39 -> 16 -> 40 -> 17 -> 39 -> 18 -> 40 -> 17 -> 39 -> 16 -> 40 -> 17 -> 39 -> 18 -> 40 -> 17 -> 39 -> 18 -> 40 -> 17 -> 39 -> 16 -> 40 -> 17 -> 18 -> 40 -> 17 -> 39 -> 16 -> 40 -> 17 -> 18 -> 40 -> 17 -> 18 -> 40 -> 17 -> 18 -> 40 -> 1 > 14 -> 40 -> 13 -> 39 -> 12 -> 40 -> 11 -> 39 -> 10 -> 40 -> 9 -> 39 -> 8 -> 40 -> 7 -> 39 -> 6 -> 40 -> 5 -> 39 -> 4 -> 40 -> 3 -> 39 -> 2 -> 40 -> 1 -> 39 -> 0 -> 38 -> 37 -> 36 -> 38 -> 35 -> 37 -> 34 -> 38 -> 33 -> 37 -> 32 -> 38 -> 31 -> 37 -> 30 -> 38 -> 31 -> 37 -> 30 -> 38 -> 29 -> 37 -> 28 -> 38 -> 27 -> 37 -> 26 -> 38 -> 25 -> 37 -> 24 -> 38 -> 23 -> 37 -> 22 -> 38 -> 21 -> 37 -> 20 -> 38 -> 19 -> 37 -> 18 -> 38 -> 17 -> 37 -> 16 -> 38 -> 15 -> 37 -> 14 -> 38 -> 13 -> 37 -> 12 -> 38 -> 11 -> 37 -> 10 -> 38 -> 9 -> 37 -> 8 -> 38 -> 7 -> 37 -> 6 -> 38 -> 5 -> 37 -> 4 -> -> 23 -> 35 -> 22 -> 36 -> 21 -> 35 -> 20 -> 36 -> 19 -> 35 -> 18 -> 36 -> 17 -> 35 -> 16 -> 16 -> 36 -> 15 -> 15 -> 14 -> 36 -> 13 -> 35 -> 12 -> 36 -> 11 -> 35 -> 10 -> 36 -> 9 -> 35 -> 8 -> 25 -> 33 -> 24 -> 34 -> 23 -> 33 -> 22 -> 34 -> 21 -> 33 -> 20 -> 34 -> 19 -> 33 -> 18 -> 18 -> 34 -> 17 -> 33 -> 16 -> 34 -> 15 -> 33 -> 14 -> 34 -> 13 -> 33 -> 12 -> 34 -> 11 -> 33 -> 10 -> 34 -> 24 -> 32 -> 23 -> 31 -> 22 -> 32 -> 21 -> 31 -> 20 -> 32 -> 19 -> 31 -> 18 -> 32 -> 17 -> 31 -> 16 -> 32 -> 15 -> 15 -> 31 -> 14 -> 32 -> 13 -> 13 -> 12 -> 32 -> 11 -> 31 -> 10 -> 32 -> 9 -> 31 -> 8 -> 32 -> 7 -> 31 -> 6 -> 32 -> 5 -> 31 -> 4 -> 32 -> 3 -> 31 -> 2 -> 32 -> 1 -> 31 -> 0 -> 30 -> 29 -> 28 -> 30 -> 27 -> 29 -> 26 -> 30 -> 25 -> 29 -> 24 -> 30 -> 23 -> 29 -> 22 -> 30 -> 21 -> 29 -> 20 -> 30 -> 19 -> 29 -> 18 -> 30 -> 17 -> 29 -> 16 -> 30 -> 15 -> 29 -> 14 -> 30 -> 13 -> 29 -> 12 -> 30 -> 11 -> 29 -> 10 -> 30 -> 9 -> 29 -> 8 -> 30 -> 7 -> 29 -> 6 -> 30 -> 5 -> 29 -> 4 -> 30 -> 3 -> 29 -> 2 -> 30 -> 1 -> 29 -> 0 -> 28 -> 27 -> 26 -> 28 -> 25 -> 27 -> 24 -> 28 -> 23 -> 27 -> 22 -> 28 -> 21 -> 27 -> 20 -> 28 -> 19 -> 27 -> 18 -> 28 -> 17 -> 27 -> 16 -> 28 -> 15 -> 27 -> 14 -> 28 -> 13 -> 27 -> 12 -> 28 -> 11 -> 27 -> 10 -> 28 -> 9 -> 27 -> 8 -> 28 -> 7 -> 27 -> 6 -> 28 -> 5 -> 27 -> 4 -> 28 -> 3 -> 27 -> 27 -> 28 -> 1 -> 27 -> 0 -> 26 -> 25 -> 24 -> 26 -> 23 -> 25 -> 22 -> 26 -> 21 -> 25 -> 20 -> 26 -> 11 -> 25 -> 10 -> 10 -> 26 -> 9 -> 25 -> 8 -> 26 -> 7 -> 25 -> 6 -> 26 -> 5 -> 25 -> 4 -> 26 -> 3 -> 25 -> 25 -> 26 -> 1 -> 25 -> 0 -> 24 -> 23 -> 22 -> 24 -> 21 -> 23 -> 20 -> 24 -> 19 -> 23 -> 18 -> 24 -> 17 -> 23 - 16 - 24 - 15 - 23 - 14 - 24 - 13 - 23 - 12 - 24 - 11 - 23 - 10 - 24 - 10 - 24 - 9 - 23 - 8 - 24 - 7 - 23 - 6 - 24 - 5 - 23 - 4 - 24 - 3 - 23 - 23 - 24 - 1 - 23 - 24 - 1 - 23 - 9 - 23 --> 22 -> 21 -> 20 -> 22 -> 19 -> 21 -> 18 -> 22 -> 17 -> 21 -> 16 -> 22 -> 15 -> 21 -> 14 -> 22 -> 13 -> 21 -> 12 -> 22 -> 11 -> 21 -> 10 -> 22 -> 9 -> 21 -> 8 -> 22 -> 7 -> 21 -> 6 -> 22 -> 5 -> 21 -> 4 -> 22 -> 3 -> 21 -> 2 -> 22 -> 1 -> 21 -> 0 -> 20 -> 19 -> 18 -> 20 -> 17 -> 19 -> 16 -> 20 -> 15 -> 19 -> 14 -> 20 -> 13 -> 19 -> 12 -> 20 -> 11 -> 19 -> 10 -> 20 -> 9 -> 19 -> 8 -> 20 -> 7 -> 19 -> 6 -> 20 -> 5 -> 19 -> 4 -> 20 -> 3 -> 19 -> 2 -> 20 -> 1 -> 19 -> 0 -> 18 -> 17 -> 16 -> 18 -> 17 -> 14 -> 18 -> 13 -> 17 -> 12 -> 18 -> 11 -> 17 -> 10 -> 18 -> 9 -> 17 -> 8 -> 18 -> 7 -> 17 -> 6 -> 18 -> 5 -> 17 -> 4 -> 18 -> 3 -> 17 -> 2 -> 18 -> 1 -> 17 -> 0 -> 16 -> 15 -> 14 -> 16 -> 13 -> 15 -> 12 -> 16 -> 11 -> 15 -> 10 -> 16 -> 9 -> 15 -> 8 -> 16 -> 7 -> 15 -> 6 -> 16 -> 5 -> 15 -> 4 -> 16 -> 3 -> 15 -> 2 -> 16 -> 1 -> 15 -> 0 -> 14 -> 13 -> 12 -> 14 -> 11 -> 13 -> 10 -> 14 -> 9 -> 13 -> 8 -> 14 -> 7 -> 13 -> 6 -> 14 -> 5 -> 13 -> 4 -> 14 -> 3 -> 13 -> 2 -> 14 -> 1 -> 13 -> 0 -> 12 -> 11 -> 10 -> 12 -> 9 -> 11 -> 8 -> 12 -> 7 -> 11 -> 6 -> 12 -> 5 -> 11 -> 4 -> 12 -> 3 -> 11 -> 2 -> 12 -> 1 -> 11 -> 0 -> 10 -> 9 -> 8 -> 10 -> 7 -> 9 -> 6 -> 10 -> 5 -> 9 -> 4 -> 10 -> 3 -> 9 -> 2 -> 10 -> 1 -> 9 -> 0 -> 8 -> 7 -> 6 -> 8 -> 5 -> 7 -> 4 -> 8 -> 3 -> 7 -> 2 -> 8 -> 1 -> 7 -> 0 -> 6 -> 5 -> 4 -> 6 -> 3 -> 5 -> 2 -> 6 -> 1 -> 5 -> 0 -> 4 -> 3 -> 2 -> 4 -> 1 -> 3 -> 0 -> 1 -> 2 -> 0



Dificuldades



- Iniciais
 - Conhecimento em teoria de grafos
- Implementação
 - Desenvolver um método de validação para aresta de corte
 - Trabalhar DFS sem recursividade
- Bugs/erros
 - Validar saída para diversos casos



Bibliografia

- https://www.inf.ufsc.br/grafos/temas/euleriano/euleriano.htm
- http://www.gpec.ucdb.br/pistori/disciplinas/discreta/aulas/geulham.htm
- https://www.youtube.com/watch?v=mRvt5yJTw7E
- http://tmrfindia.org/sutra/v2i21.pdf
- https://www.ic.unicamp.br/~meidanis/research/fa/
- https://www.geeksforgeeks.org/eulerian-path-and-circuit/
- https://www.tutorialspoint.com/fleury-s-algorithm-for-printing-eulerian-path-or-circuit-in-cplusplus
- https://www.tutorialspoint.com/Fleury-s-Algorithm
- https://www.geeksforgeeks.org/implementation-of-bfs-using-adjacency-matrix/





THANK YOU!

