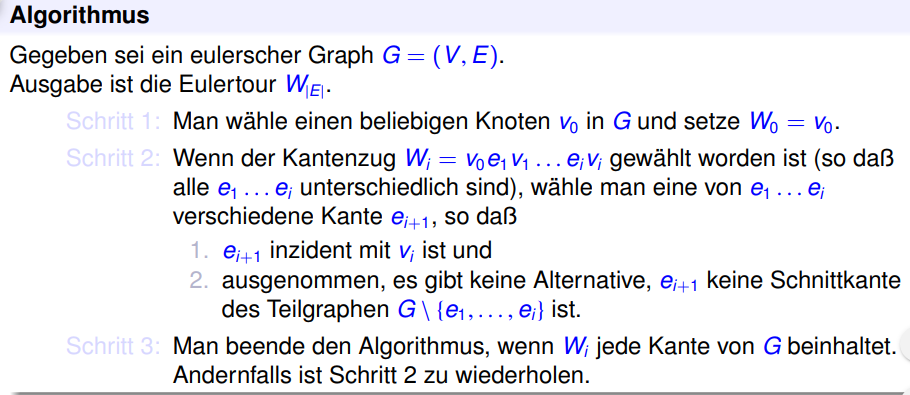
# Fleury Algorithmus

Der Fleury-Algorithmus wird verwendet, um den EulerTour (Euler Zyklus) aus einem gegebenen Graphen anzuzeigen. Bei diesem Algorithmus wird ausgehend von einer Kante versucht, andere benachbarte Knoten zu verschieben, indem die vorherigen Kanten entfernt werden. Mit diesem Trick wird es in jedem Schritt einfacher, den Euler-Pfad oder die Euler-Zyklus zu finden.



**Implementierung**

Wir finden zuerst den Startpunkt, der ein zufälliger Vertex u sein kann und speichern ihn in der Variablen currentNode. Um einen zufälligen Knoten zu erhalten, erhalten wir den Knoten mit einem zufälligen Index im Diagramm, solange der Index kleiner als die Anzahl der Knoten im Diagramm ist.

Wir speichern den Euler-Kreis in einer LinkedList und geben dann die Euler-Tour zurück, beginnend mit dem Startknoten u in der Methode getEulerCircle ().

Alle Kanten im Diagramm werden in der Liste EdgeList gespeichert.

Bevor wir den Fleury-Algorithmus ausführen, müssen wir die Bedingungen des Graphen überprüfen. Es muss diese Dinge erfüllen: Der gesamte Knotengrad muss gerade sein und das Diagramm ist ungerichtet, der aktuelle Knoten ist nicht null und die Kantenliste ist nicht leer. Die Eigenschaften einer EulerTour wird in der Klasse EulerCycleProperties geschrieben.

Der Algorithmus wird in folgenden Schritten programmiert:

Der Algorithmus fügt eine anfänglich leere Kantensequenz hinzu und erstellt einen Eulerkreis

1. Wählen Sie einen beliebigen Knoten als aktuellen Knoten aus.

2. Wählen Sie die Kante mit dem aktuellen Knoten in Presidents. In diesem Fall müssen zuerst Kanten ausgewählt werden, die keine überbrückten Kanten im Diagramm sind.

3. Löschen Sie die ausgewählte Kante im Diagramm und fügen Sie sie dem Eulerkreis (der Liste) hinzu.

4. Wählen Sie den anderen Knoten der ausgewählten Kante als neuen aktuellen Knoten.

5. Wenn das Diagramm Kanten enthält, fahren Sie mit Schritt 2 fort.

Eine Kante in einem ungerichteten verbundenen Diagramm ist eine Brücke, wenn durch Entfernen die Verbindung zum Diagramm getrennt wird.

Die Methode isBridge () dient dazu, die Brücken im Diagramm nacheinander zu suchen, um alle Kanten zu entfernen und festzustellen, ob das Entfernen einer Kante zu einem nicht verbundenen Diagramm führt.

Die Methode wird in folgenden Schritten programmiert:

 Für jede Kante:

  1. Kante vom Graphen entfernen

  2. Überprüfen Sie, ob der Graph weiterhin verbunden ist. (Wir verwenden BFS, um zu überprüfen, ob wir den BFS-Pfad von finden können

der aktuelle Knoten zurück zum Startknoten

3. Fügen Sie dem Diagramm eine Kante hinzu.

**Test**

Wir implementieren folgende Tests:

Negative Test: Wir prüfen die Eingenschaften von Euler Zyklus in dem Graph.

Positive Test: Die Anzahl von Kanten in der Kreis ist gleich als die Anzahl von Kanten in der Graph