

## Drittes Übungsblatt

**Ausgabe:** 27. Mai 2020

**Besprechung:** 9. Juni 2020

### 1 Triangulierung

1. Geben Sie einen *linearen* (in der Anzahl der Knoten  $n$ ) Algorithmus an, der eine Triangulierung  $G' = (V, E')$  von  $G$  mit kombinatorischer Einbettung  $\mathcal{G}'$  findet.  
Hinweis: Die Triangulierung muss einfach sein, darf also insbesondere keine Mehrfachkanten enthalten.
2. Führen Sie Ihren Algorithmus an folgenden Beispielgraphen aus und numerieren Sie jeweils die von Ihrem Algorithmus eingefügten Kanten in der Reihenfolge, in der Ihr Algorithmus sie einfügt.

a)  $K_{1,2}$       b)  $K_{1,3}$       c)  $Q_2$       d)  $G_1$

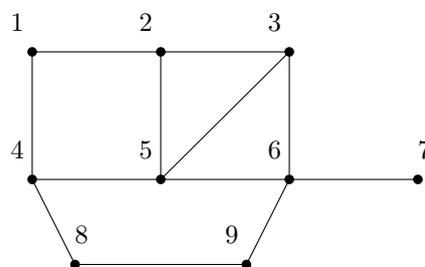


Abbildung 1: Der Graph  $G_1$  zu Aufgabe 1

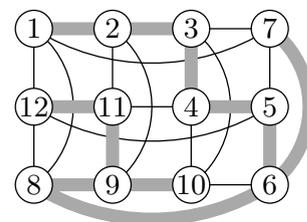
### 2 Minoren vs topologische Minoren

1. Zeigen Sie auf drei verschiedene Arten, dass der Petersengraph nicht planar ist.
2. Zeigen oder widerlegen Sie: Wenn ein einfacher Graph  $H$  eine Unterteilung eines einfachen Graphen  $G$  als Teilgraph enthält, dann enthält  $H$  den Graphen  $G$  auch als Minor.
3. Zeigen oder widerlegen Sie: Wenn ein einfacher Graph  $H$  einen Graphen  $G$  als Minor enthält, dann enthält  $H$  auch eine Unterteilung von  $G$  als Teilgraph.

**Bitte wenden**

### 3 LR-Planarität

1. Testen Sie ob der gegebene Graph planar ist, indem Sie das LR-Planaritätskriterium bezüglich des eingezeichneten DFS-Baumes (fette Kanten) überprüfen.
2. Beschreiben Sie grob einen einfachen Algorithmus mit polynomieller Laufzeit der Planarität mithilfe des LR-Planaritätskriteriums prüft.



### 4 Umfang

Der *Umfang* (engl. girth) eines Graphen  $G$  ist die Länge eines kürzesten Kreises in  $G$ . Enthält  $G$  keinen Kreis, so ist der Umfang  $\infty$ .

- a) Geben Sie einen Algorithmus an, der für einen gegebenen Knoten  $v$  von  $G$  entweder
  - die Länge des kürzesten Kreises berechnet auf dem  $v$  liegt, oder
  - entscheidet, dass  $v$  nicht auf einem Kreis in  $G$  liegt.
- b) Verwenden Sie das Verfahren aus Aufgabenteil a), um für einen beliebigen Graphen den Umfang zu berechnen. Welche Laufzeit erhalten Sie?
- c) Beschleunigen Sie Ihren Algorithmus für den Fall, dass der Eingabegraph planar ist.

### 5 Adjazenztest in planaren Graphen

Sei  $G$  ein planarer Graph mit  $n$  Knoten. Geben Sie eine Datenstruktur mit linearer Größe an, mit deren Hilfe nach linearer Vorberechnung Adjazenzen von Knoten in konstanter Zeit abgefragt werden können. Das heißt, gegeben zwei Knoten  $u$  und  $v$  von  $G$ , kann die Frage ob die Kante  $\{u, v\}$  in  $G$  ist, in konstanter Zeit beantwortet werden.