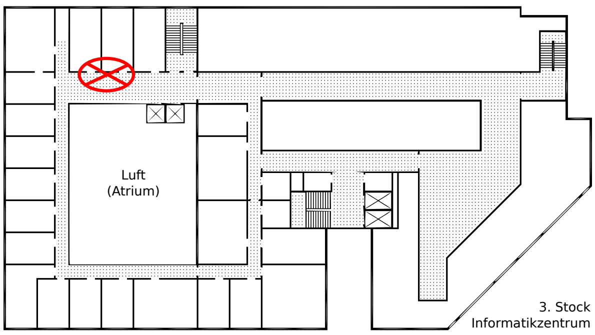
Abteilung Algorithmik
Institut für Betriebssysteme und Rechnerverbund
TU Braunschweig

Winter 2015/16

Prof. Dr. Sándor P. Fekete
Dr. Christian Scheffer
Jan-Marc Reinhardt

Algorithmen und Datenstrukturen
Übung 1 vom 11. 11. 2015

Abgabe der Lösungen bis zum Mittwoch, den 25. 11. 2015 um 13:00 im Hausaufgabenrückgabeschrank.



Bitte die Blätter zusammenheften und vorne deutlich mit eigenem Namen, Matrikel- und Gruppennummer, sowie Studiengang versehen!

Aufgabe 1 (Eulertouren und Hamiltonkreise): Betrachte die Graphen P_1 und P_2 aus Abbildung 1. So viel sei verraten: Beide enthalten keinen Hamiltonkreis.

a) Wie viele Knoten (einschließlich inzidenter Kanten) muss man mindestens aus P_2 entfernen, damit der resultierende Graph einen Hamiltonkreis enthält? Begründe deine Antwort!

b) Enthält P_1 eine Eulertour? Falls ja, gib eine Eulertour in P_1 an. Andernfalls füge möglichst wenige Kanten zu P_1 hinzu, sodass der resultierende (einfache!) Graph P_1' eine Eulertour enthält, und gib eine Eulertour in P_1' an.

(5+10 Punkte)

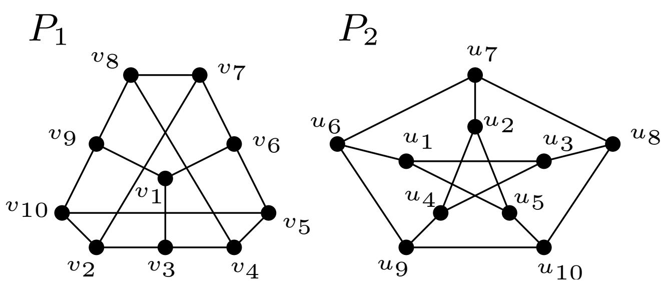


Abbildung 1: Die Graphen P_1 und P_2

Aufgabe 2 (Graphen und Einbettungen): Ein Graph heißt *planar*, falls man ihn so zeichnen kann, dass sich keine zwei Kanten kreuzen.[1]

[1]Formaler: falls er eine kreuzungsfreie Einbettung in die Ebene hat.

a) Ist der Graph K_4 aus Abbildung 2 planar? Begründe deine Antwort!

b) Der Graph K_5 aus Abbildung 2 ist bekannt dafür, nicht planar zu sein. Zeige: Entfernt man eine beliebige Kante e aus dem K_5, so ist der resultierende Graph $K_5 \setminus \{e\}$ planar.

(5+10 Punkte)

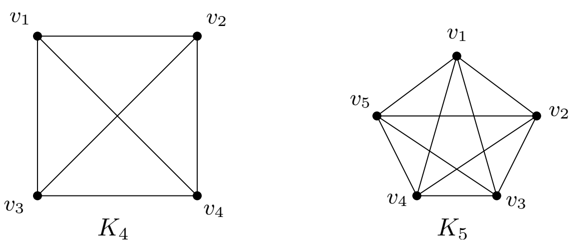


Abbildung 2: Die Graphen K_4 und K_5

Aufgabe 3 (Graphenisomorphie): Wir betrachten einen abgeschwächten Gleichheitsbegriff für Graphen, die Graphenisomorphie.[2] Zwei einfache Graphen $G_1 = (V_1, E_1)$ und $G_2 = (V_2, E_2)$ sind *isomorph*, wenn es eine bijektive Abbildung $p : V_1 \to V_2$ gibt, die adjazenzerhaltend ist, d.h., $\{v_i, v_j\} \in E_1 \Leftrightarrow \{p(v_i), p(v_j)\} \in E_2$.

Die *Knotengradfolge* eines Graphen $G = (V, E)$ ist eine sortierte Folge $\delta(v_{i_1}) \leq \delta(v_{i_2}) \leq \ldots \leq \delta(v_{i_n})$ mit $V = \{v_{i_1}, v_{i_2}, \ldots, v_{i_n}\}$ und $|V| = n$, wobei $\delta(v)$ der Grad eines Knotens v ist, d.h. die Anzahl seiner inzidenten Kanten.

a) Zeige: Die Graphen P_1 und P_2 aus Abbildung 1 sind isomorph.

b) Sind die Graphen G und H aus Abbildung 3 isomorph? Begründe deine Antwort!

c) Zeige: Haben zwei Graphen G_1 und G_2 eine unterschiedliche Knotengradfolge, dann sind sie nicht isomorph.

d) Vervollständige die Zeilen 3, 4 und 7 in Algorithmus 1, der bestimmt, ob zwei Knotengradfolgen identisch sind. Dabei kannst du auf den i-ten Knotengrad der ersten Folge mit v_i zugreifen und auf den i-ten Knotengrad der zweiten Folge mit u_i. Sind die Knotengradfolgen gleich, soll der Algorithmus `true` zurückgeben, sonst `false`. Du kannst davon ausgehen, dass beide Folgen die Länge n haben.

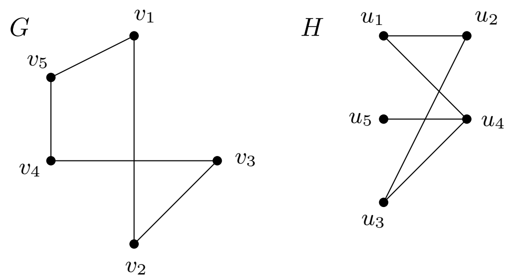


Abbildung 3: Die Graphen G und H

(10+5+10+5 Punkte)

[2] Auf die Definition der Graphenisomorphie wird in der 2. großen Übung am 19.11. näher eingegangen.

```
function GLEICHE_KNOTENGRADFOLGE([v], [u])
    for i ← 1, . . . , n do
        if                                  then

        end if
    end for

end function
```

Algorithmus 1: Algorithmus zum Vergleich zweier Knotengradfolgen