Algorithmen und Datenstrukturen

Große Übung vom 04.11.10

Christiane Schmidt

Diese Folien

- Braucht man nicht abzuschreiben
- Stehen im Netz unter

  www.ibr.cs.tu-bs.de/courses/ws1011/aud/index.html

Kleine Übungen

- Anmeldung online auf

  www.ibr.cs.tu-bs.de/courses/ws1011/aud/index.html
- Ca. 20 Teilnehmer pro Übung
- Das Online Anmeldeverfahren gibt Auskunft über den Erfolg der Anmeldung
  - Sie haben sich erfolgreich angemeldet
  - Jeder der diese Meldung bekommen hat, ist in der entsprechenden Gruppe
- Probleme?

Kleine Übungen

Mittwoch

- 13:15 - 14:45 Uhr, PK 3.1, PK 3.2
- 15:00 - 16:30 Uhr, BW 74.3, BW 74.4
- 16:45 - 18:15 Uhr, BW 74.3, BW 74.4

Donnerstag

- 09:45 – 11:15 Uhr, RR 58.3, LK 19a.1
- 13:15 - 14:45 Uhr, RR 58.3, RR 58.4
- 15:00 - 16:30 Uhr, BW 74.3, BW 74.4

Freitag

- 09:45 - 11:15 Uhr, BW 74.6, BW 74/131

Mailingliste

- Natürlich auf: www.ibr.cs.tu-bs.de/courses/ws1011/aud/index.html
- Name + Emailadresse + Passwort eintragen und absenden
- Mailingliste schickt E-Mail an die angegebene Adresse
- E-Mail der Mailingliste per „Antworten“ zurückschicken. Fertig!

Hausaufgaben

- 5 Hausaufgabenblätter
- 14-tägig auf www.ibr.cs.tu-bs.de/courses/ws1011/aud/index.html
- Bearbeitungszeit: 14 Tage
- Abgabe: Mittwochs bis 11:25 vor dem IZ 262

Hausaufgaben

Hausaufgaben

- 5 Hausaufgabenblätter
- 14-tägig auf www.ibr.cs.tu-bs.de/courses/ws1011/aud/index.html
- Bearbeitungszeit: 14 Tage
- Abgabe: Mittwochs bis 11:25 vor dem im IZ 262
- Rückgabe: In den kleinen Übungen
- 50% der Hausaufgabenpunkte sind Voraussetzung für Klausurteilnahme
  (für alle Bachelor- / Masterstudenten/innen)
- Besprechung der Lösungen in den kleinen Übungen

Große Übung

- Aufarbeitung des Vorlesungsstoffes
- An manchen Stellen auch Vertiefung des Stoffes
- Ihr könnt Themen mitbestimmen! Dazu einfach eine E-Mail an mich.
- Fragen stellen ausdrücklich erlaubt!

Übersicht über das Semester

KW	VL - Nr. (Di+Mi)	Gr. UE (Do)	Kl. UE (Mi+Do+Fr)	HA Ausgabe (Mi abends)	HA Abgabe (Mi bis 11:25 Uhr)	HA Rückgabe (in kl. UE)
43	-	-	-			
44	1, 2	1		UE_0 *		
45	3, 4			UE_0B *		
46	5, 6	2	1	HA_1		UE_0 *
47	7, 8					
48	9, 10	3	2	HA_2	HA_1	UE_0B *
49	11, 12					
50	13, 14	4	3	HA_3	HA_2	HA_1
51	Weihnachtsferien					
52						
1	15, 16	5	4	HA_4	HA_3	HA_2
2	17,18					
3	19, 20	6	5	HA_5	HA_4	HA_3
4	21, 22					
5	23, 24	7	6		HA_5	HA_4
6	25, 26		7			HA_5

*Bearbeitung in den ersten kleinenen Übungen; keine Bewertung

Klausur

- Termin: 22.02.2011

Das Rundreise-Problem

Fortsetzung der Rundreise

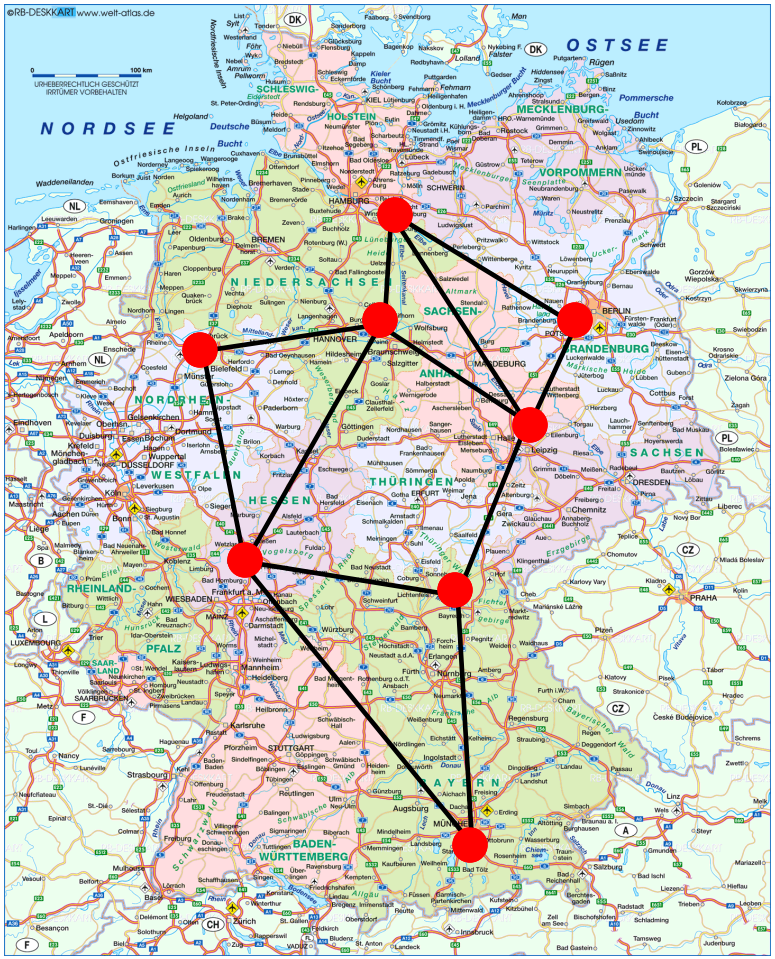
- Problem: Finde eine Rundreise, (d.h. eine Route die jeden Ort besucht und zum Startort zurückkehrt) die möglichst kurz ist.

Fortsetzung der Rundreise

Fortsetzung der Rundreise

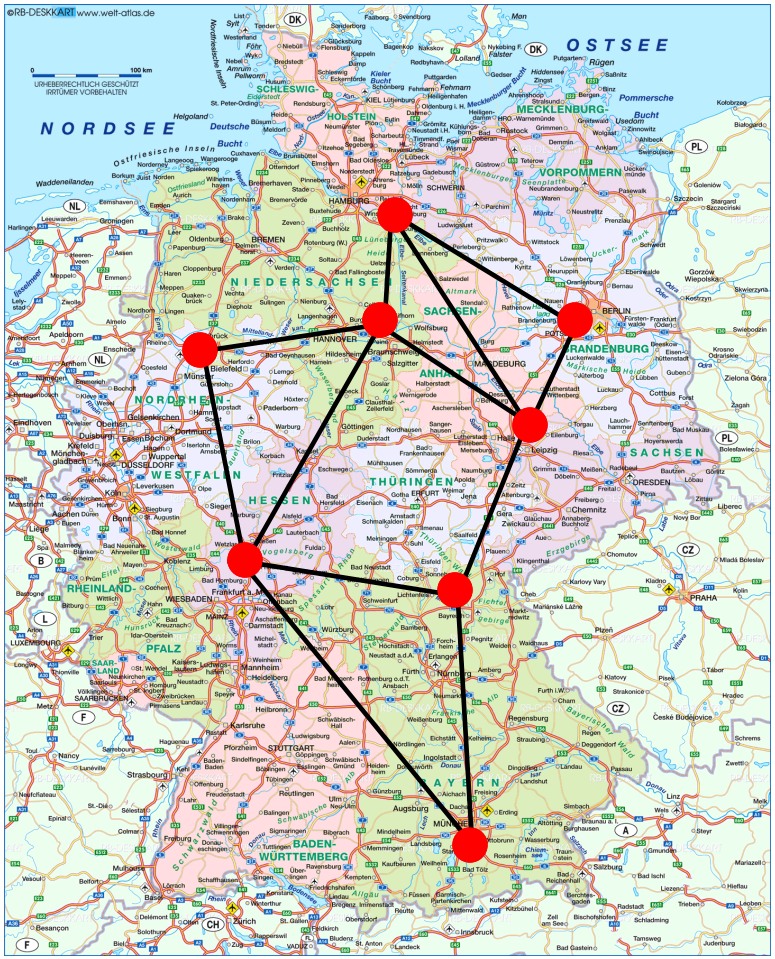
- Wie funktioniert map24.de?
- Man kann zwischen den Städten beliebig hin und her reisen.

Phase 1: Modellierung



- Zur Modellierung kann man Graphen benutzen
- Ein Graph *G* besteht aus Knoten ● und Kanten —

Phase 1: Modellierung

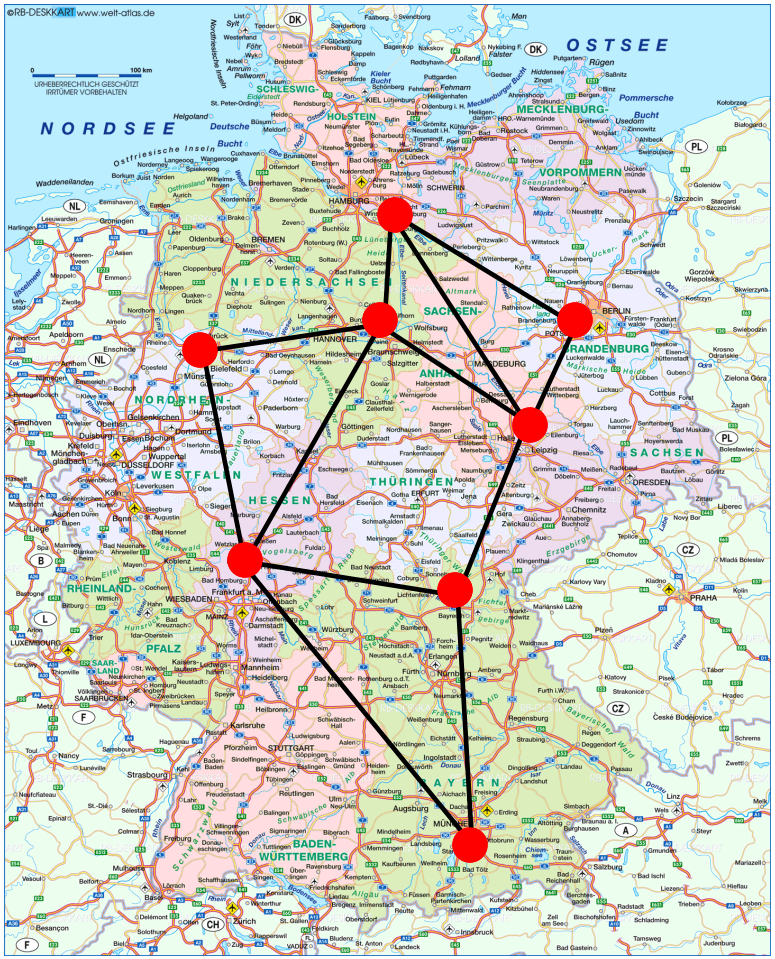


- Knoten:
  - Hamburg
  - Berlin
  - Braunschweig
  - Leipzig
  - ……

Typische Schreibweise

- Kanten:
  - {Hamburg, Berlin}
  - {BS, Berlin}
  - {BS, Leipzig}
  - ……

Phase 2: Datenstruktur



- Daten in „geeigneter Form“ an den Computer übergeben
- Was sind die relevanten Daten in dem Problem?
  - Städte
  - Verbindungen
  - Entfernungen

Phase 2: Datenstruktur

- Eine anwendbare Datenstruktur sind z.B. Listen
- Speichere alle „benachbarten Orte“ in einer Liste:
  (Hamburg, Leipzig, Münster, Frankfurt,…)
- Die Entfernungen in einer anderen:
  - BS_Ent:
    (220, 200, 250,…)

Bald in dieser Vorlesung

Phase 3: Algorithmen

- Es ist „schwierig“ die beste Lösung zu finden
- Aber man kann schnell „zi[illegible]ute“ finden
  [illegible]ithmus:
  - Wähle bel. Startort
  - Gehe zum nächstgelegenen Ort außer, dieser ist bereits besucht

Bald in dieser Vorlesung

Turnierplanung

Szenario

- n Spieler einer Mannschaft sollen nach Spielstärke aufgestellt werden
- Dazu: Turnier „jeder gegen jeden“
- Pro Abend spielt jeder Spieler ein Spiel
- Frage: wie viele Abende werden benötigt?

Anzahl

- Sei n=6 (wir haben 6 Spieler)
- Jeder Spieler muss genau einmal gegen jeder der 5 anderen Spieler antreten
- Damit: 6*5/2 = 15 Spiele zu absolvieren (denn: Spiel i gegen j würde für beide Spieler gezählt werden → *1/2)
- Da jeder Spieler ein Spiel am Abend: 3 Spiele pro Abend
- → 15/3=5 Abende werden benötigt

Erster Ansatz: einfach Spielen

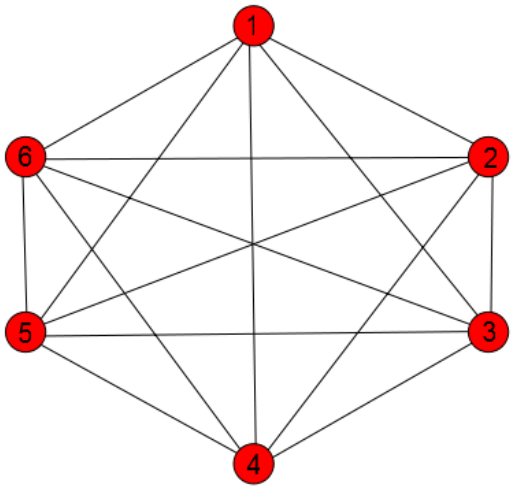
1. Abend	2. Abend	3. Abend
1-2	1-3	1-4
3-5	2-6	2-5
4-6	4-5	3-6

Aber:
restlichen 6 Spielpaarungen nicht an 2 Abenden!!
(1-5 kann weder mit 1-6 noch 5-6 parallel ausgetragen werden)

4. Abend	5. Abend	6. Abend
1-5		5-6
2-3	2-4	3-4

Hier muss auch je ein Spieler aussetzen!!

- Wir modellieren das Problem mit Hilfe eines Graphen:
  - Knoten: jeder Spieler
  - Kanten: Spiele
- Für n=6:

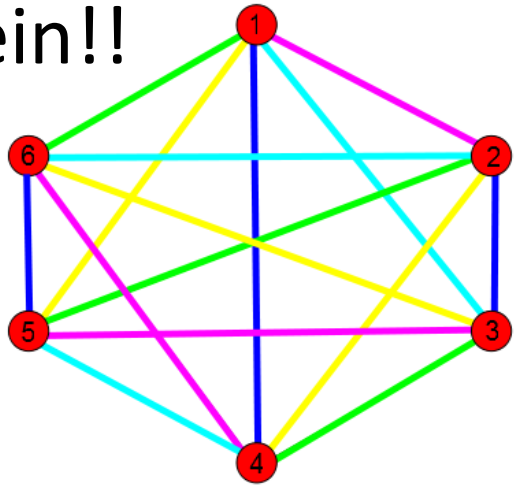


Graphen in denen jeder Knoten mit jedem verbunden ist heißen vollständig

Ansatz: Färben

- Wir färben die Kanten mit den Farben 1,…, n-1
- Jede Farbe entspricht einer Runde
- Färbung zulässig – wenn sie zulässigem Spielplan entspricht

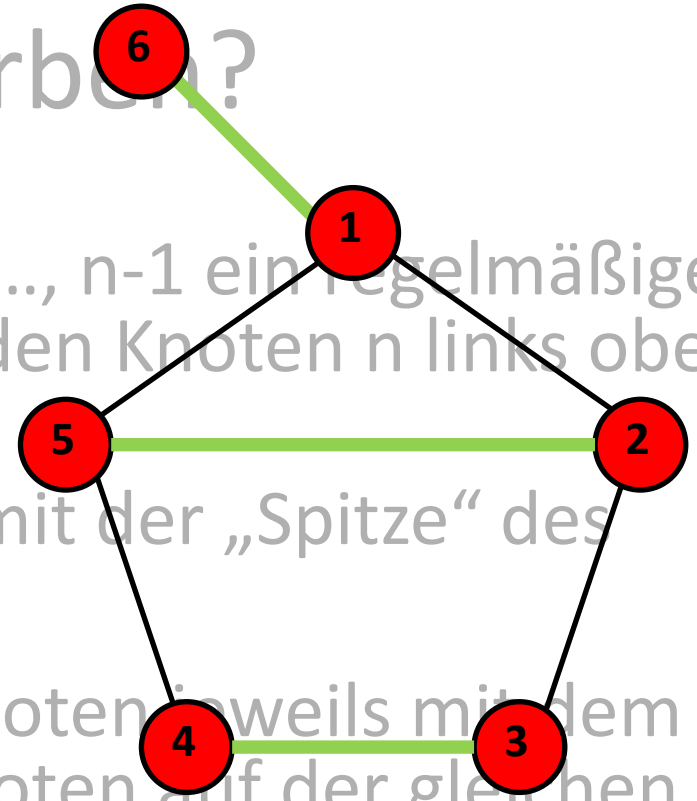
➢ Alle Kanten, die in gleich Knoten führen, müssen unterschiedlich gefärbt sein!!



Wie Färben?

1. Bilde aus den Knoten 1,…, n-1 ein regelmäßiges (n-1)-Eck und platziere den Knoten n links oben neben dem (n-1)-Eck
2. Verbinden den Knoten mit der „Spitze“ des (n-1)-Ecks
3. Verbinde die übrigen Knoten jeweils mit dem gegenüberliegenden Knoten auf der gleichen Höhe im (n-1)-Eck
4. Die eingefügten n/2 Kanten werden mit der ersten Farbe gefärbt

Wie Färben?

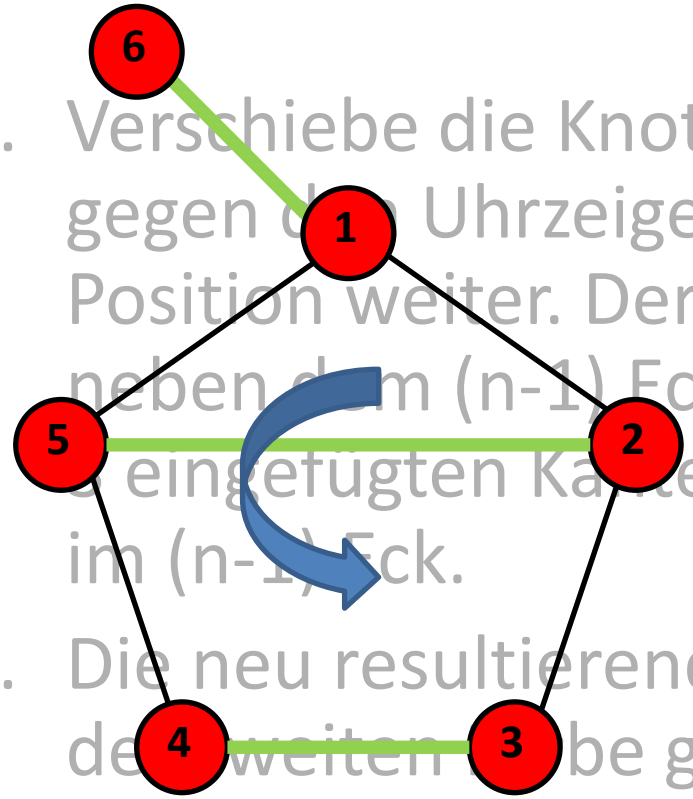


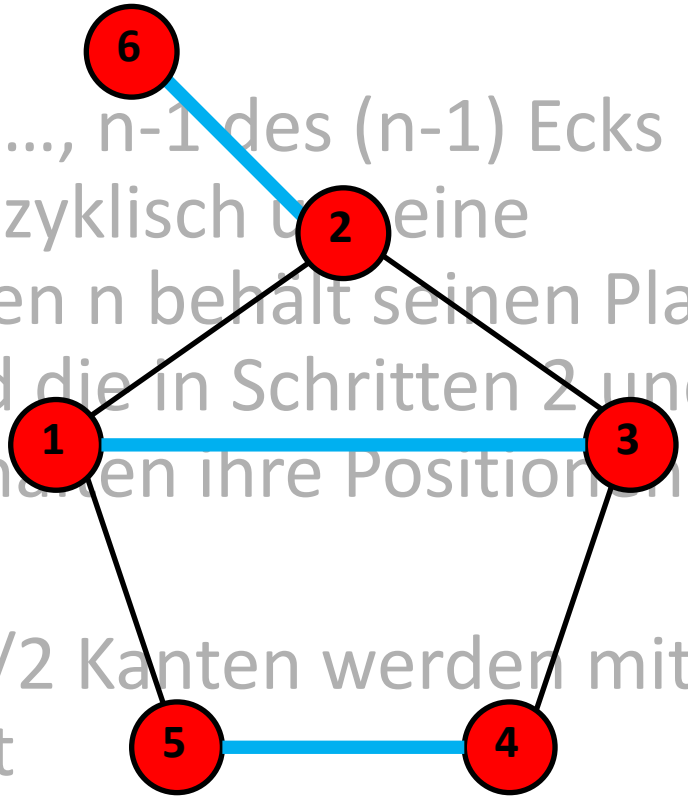
1. Bilde aus den Knoten 1,…, n-1 ein regelmäßiges (n-1)-Eck und platziere den Knoten n links oben neben dem (n-1)-Eck
2. Verbinden den Knoten mit der „Spitze“ des (n-1)-Ecks
3. Verbinde die übrigen Knoten jeweils mit dem gegenüberliegenden Knoten auf der gleichen Höhe im (n-1)-Eck
4. Die eingefügten n/2 Kanten werden mit der ersten Farbe gefärbt

Wie Färben? (Fortsetzung)

5. Verschiebe die Knoten 1, …, n-1 des (n-1) Ecks gegen den Uhrzeigersinn zyklisch um eine Position weiter. Der Knoten n behält seinen Platz neben dem (n-1) Eck, und die in Schritten 2 und 3 eingefügten Kanten behalten ihre Positionen im (n-1) Eck.
6. Die neu resultierenden n/2 Kanten werden mit der zweiten Farbe gefärbt
7. Die Schritte 5 und 6 werden für die Farben 3, …, n-1 wiederholt

Wie Färben? (Fortsetzung)





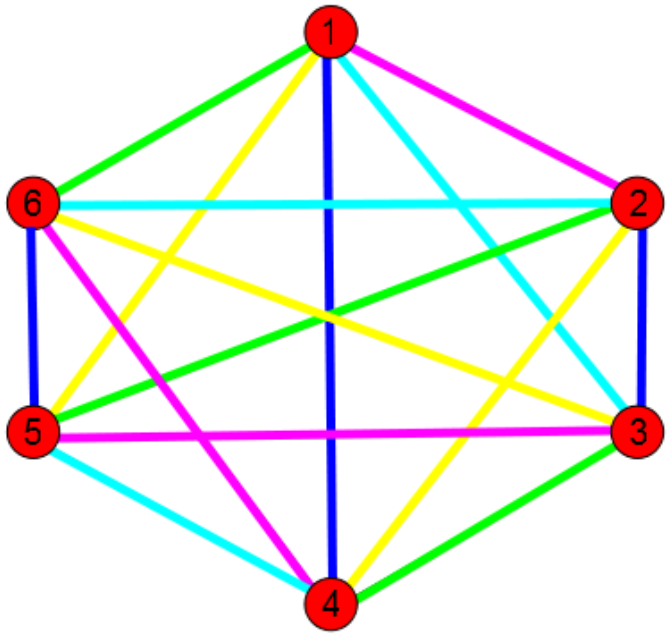
5. Verschiebe die Knoten 1, …, n-1 des (n-1) Ecks gegen den Uhrzeigersinn zyklisch um eine Position weiter. Der Knoten n behält seinen Platz neben dem (n-1) Eck, und die in Schritten 2 und 3 eingefügten Kanten behalten ihre Positionen im (n-1) Eck.
6. Die neu resultierenden n/2 Kanten werden mit der zweiten Farbe gefärbt
7. Die Schritte 5 und 6 werden für die Farben 3, …, n-1 wiederholt

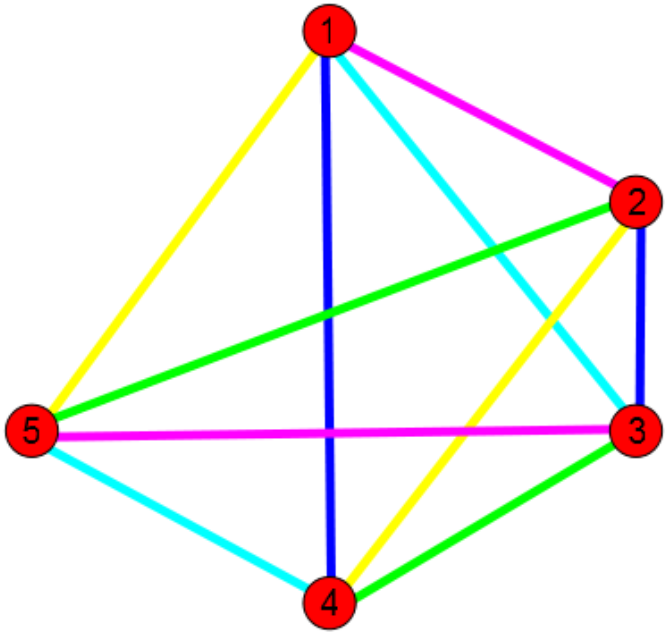
usw.

Wieso funktioniert dieser Algorithmus?

- Beobachtung:

Entfernen wir Knoten 6 und alle 5 seiner Kanten, erhalten wir 5-Eck, in dem alle 5 Kanten auf dem Rand unterschiedlich gefärbt sind.

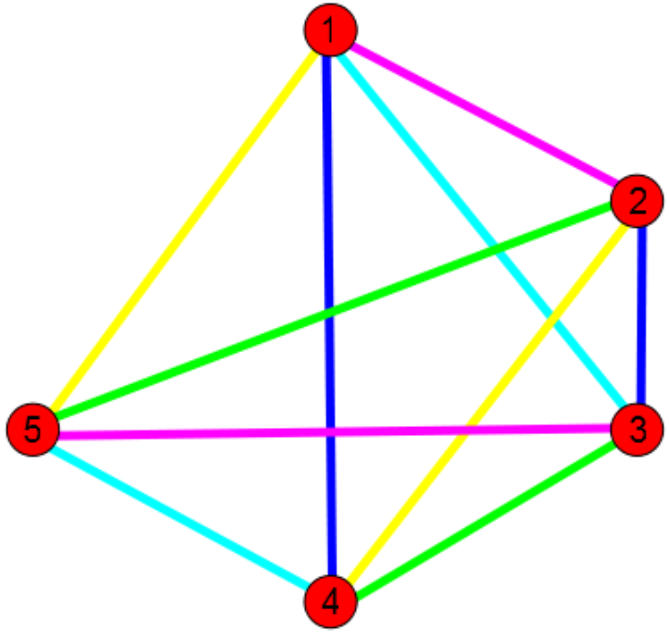


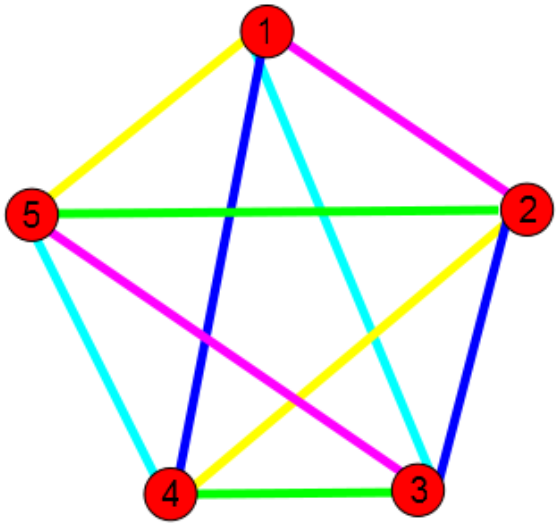


Wieso funktioniert dieser Algorithmus?

- Beobachtung 2:

Betrachten wir dies als regelmäßiges 5-Eck, fällt auf, dass jede Kante im Inneren die gleiche Farbe hat, wie die parallele Kante auf Rand.





Wieso funktioniert dieser Algorithmus?

- Wenn immer nur parallele Kanten die gleiche Farbe haben, die ja nicht in den gleichen Knoten führen, ist unsere Zulässigkeitsbedingung erfüllt
- Und: an jedem der 5 Knoten werden 4 Farben verwendet – dabei jeweils eine andere, die für die Verbindung zu Knoten 6 verwendet werden kann

➢ Funktioniert für jeden Graphen mit gerader Anzahl n an Knoten

Einfachere Formulierung

- Oben: wir müssten (n-1)-Ecke speichern
- Geht das einfacher?
- Dazu: Modulo-Rechnung (Division mit Rest)

Division mit Rest

- Zwei natürliche Zahlen *a* und *b* sollen mit Rest dividiert werden, d.h., man sucht eine Darstellung: $a = b*n + r$
- Die Modulofunktion ordnet zwei Zahlen den Teilerrest zu – also: $a \bmod b = r$
- Beispiele:
  - 19 mod 7 = 5,
  - denn: 19=2*7+5
  - 37 mod 17 = 3,
  - denn: 37= 2*17 + 3

Einfachere Formulierung

1. Für alle Farben
2. Färbe die Kante [i,n] mit der Farbe i
3. Für k=1, …, n/2-1

   färbe alle Kanten

   [(i+k) mod (n-1), (i-k) mod (n-1)] mit

   der Farbe i.

Da unsere Knoten von 1,...,n-1 durchnummeriert sind (und nicht von 0,1,...,n-2), wird der Rest 0 als n-1 interpretiert.

Was haben wir gesehen?

- Es lohnt sich über Algorithmen nachzudenken (statt (n-1)-Ecke mit zyklischer Verschiebung, Modulo-Rechnung)
- Pseudo-Code:

```
1.  Für alle Farben
2.      Färbe die Kante [i,n] mit
        der Farbe i
3.      Für k=1, …, n/2
        färbe alle Kanten
        [(i+k) mod (n-1), (i-k) mod (n-1)] mit
        der Farbe i.
```

Schönes Wochenende….

… nächste Übung in 14 Tagen