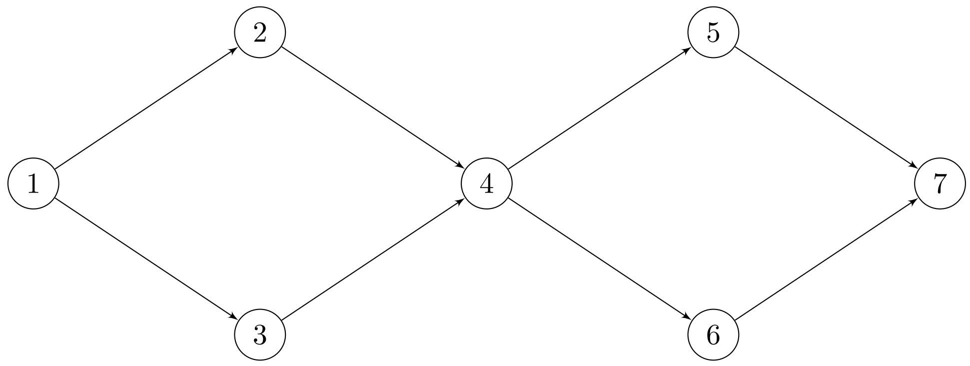
1) Für verschiedene u-v-Pfade gilt: sind sie intern-knotendisjunkt, so sind sie kantendisjunkt.

Wenn es eine gemeinsame Kante auf den Pfaden gibt, so muss auch mindestens ein Knoten gemeinsam sein. Umgekehrt gilt die Aussage nicht. Betrache die Pfade 1-2-4-5-7 und 1-3-4-6-7

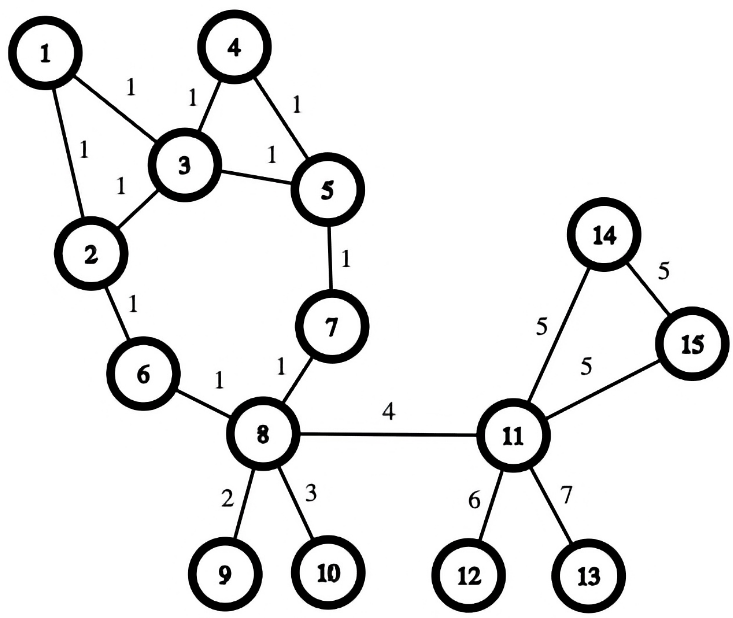


2) Ein Graph kann nie 3-kantenzusammenhängend sein, falls er

Erinnere dich an die Ungleichung: Knoten-Zsmhg. ≤ Kanten-Zsmhg. ≤ Minimaler Grad

3) Wie viele Blöcke hat der folgende Graph?

Blöcke ist eine Äquivalenzrelation auf Kanten, wobei 2 Kanten im gleichen Block sind, wenn es einen Kreis durch die beiden Kanten gibt (es können einzeln Blöcke aus nur einer Kante sein). Somit ergeben sich folgende 7 Blöcke:



4) Wie schnell können wir die Artikulationsknoten eines zusammenhängenden Graphs mit m Kanten bestimmen? (beste Schranke)

DFS mit low-Zahlen in $\mathcal{O}(m)$.

5) Hamiltonsche Graphen sind automatisch auch eulersch.

Nein, z.B. gibt es in dem kompletten Graphen auf 4 Knoten K_4 keine Eulertour, weil alle Knoten einen ungeraden Grad haben.

6) Falls ein Matching keine augmentierenden Pfade besitzt, so ist es inklusionsmaximal.

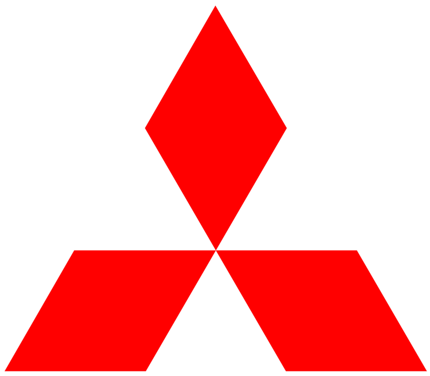
Satz von Menger. Wenn es keinen augmentierenden Pfad gibt, so ist das Matching kardinalitätsmaximal und by extension auch inklusionsmaximal.

7) Falls M ein inklusionsmaximales Matching der Grösse 8 ist, wie gross kann ein kardinalitätsmaximales Matching sein?

Abschätzung mit $|M_{ink}| \geq \frac{|M_{kar}|}{2}$.

8) Ein eulerscher Graph mit gerader Knotenanzahl hat immer ein perfektes Matching.

Nein. Bei einer Eulertour können die Knoten mehrmals vorkommen. Consider the Mitsubishi logo:



9) Sei $G = (V, E)$ ein Graph. Falls es ein $X \subseteq V$ gibt mit $|\mathcal{N}(X)| < |X|$, besitzt G kein perfektes Matching.

Ja, weil es dann nicht genug Nachbarn gibt, um jeden Knoten aus X einen Match zuzuweisen.

10) Jeder Baum ist bipartit.

Es hat keine ungeraden Kreise, somit kann man ihn mit BFS mit 2 Farben färben.

11) Wann existiert für einen Graphen G eine Knotenreihenfolge mit der der Greedy-Algorithmus nur $\chi(G)$ viele Farben braucht?

Es gibt immer solch eine Reihenfolge. Man kann sie folgendermassen konstruieren, wenn man eine Färbung von G mit $\chi(G)$ viele Farben hat: zuerst alle Knoten der Farbe 1, dann der Farbe 2 usw. Somit braucht der Greedy Algorithmus genau $\chi(G)$ viele Farben. Allerdings wissen wir nicht, wie man solch eine Reihenfolge effizient finden kann.

12) Alle zusammenhängenden Graphen mit 37 Knoten und 59 Kanten sind mit $\Delta(G)$ Farben färbbar.

Satz von Brooks. Ein Kreis auf 37 Knoten hätte 37 Kanten und ein Kompletter Graph auf 37 Knoten $\frac{37 \cdot 36}{2}$ Kanten, was viel grösser als 59 ist.

13) Es gibt 4 verschiedene Rosen. Wir wollen davon 2 für einen Blumenstrauss wählen, wie viele Möglichkeiten gibt es dafür?

Wir wählen die Rosen ungeordnet und ohne Zurücklegen, also haben wir $\binom{4}{2} = \frac{4!}{2!\cdot(4-2)!} = 6$ Möglichkeiten.

14) Sei G ein 3-färbbarer Graph. Dann können wir für jedes v den Nachbarschaftsgraphen $G[\mathcal{N}(v)]$ mit zwei Farben färben.

Alle Knoten aus $\mathcal{N}(v)$ sind Nachbarn mit v und können somit nicht die gleiche Farbe wie v haben. Also bleibt es für sie nur 2 Farben übrig.

15) Sei $\Pr[A \cup B] = 0.8$, $\Pr[A \cap B] = 0.2$ und $\Pr[A] = 0.5$. Dann ist $\Pr[B] =$

$\Pr[A \cup B] = \Pr[A] + \Pr[B] - \Pr[A \cap B]$, somit ist $\Pr[B] = 0.5$

16) Für Ereignisse A und B mit $\Pr[B] > 0$, wie ist die bedingte Wahrscheinlichkeit $\Pr[A|B]$ definiert?

$\Pr[A|B] = \frac{\Pr[A\cap B]}{\Pr[B]}$

17) Seien $(A_i), i \in [3]$ Ereignisse. Falls definiert, was ist $\Pr[A_1] \cdot \Pr[A_2|A_1] \cdot Pr[A_3|A_1 \cap A_2]$?

Multiplikationssatz.

18) Seien A, B zwei Ereignisse mit $\Pr[A] = 0.25$, $\Pr[B] = 0.5$ und $\Pr[A \cup B] = 0.625$. A und B sind unabhängig.

$\Pr[A \cup B] = \Pr[A] + \Pr[B] - \Pr[A \cap B]$, somit ist $\Pr[A \cap B] = 0.125 = \Pr[A] \cdot \Pr[B]$

19) Sind A, B, C paarweise unabhängig, so sind sie auch zu dritt unabhängig.

Beispiel 2.20 aus dem Skript, Seiten 104 und 105.

20) Seien X und Y Zufallsvariablen so dass $X \sim Bin(20, 0.2)$ und $(X + Y) \sim Geo(0.1)$ Was ist $\mathbb{E}[Y]$?

$\mathbb{E}[X] = 20 \cdot 0.2 = 4$, $\mathbb{E}[X + Y] = \frac{1}{0.1} = 10 = \mathbb{E}[X] + \mathbb{E}[Y]$, somit ist $\mathbb{E}[Y] = 6$

21) Sei F_X die Verteilungsfunktion einer Zufallsvariable. Dann gilt:

$\Pr[X \leq x]$ wird immer grösser und hat den Wert 1 im Limit.

22) Seien X und Y Zufallsvariablen. Falls $\mathbb{E}[XY] = \mathbb{E}[X]\cdot E[Y]$, sind X und Y unabhängig.

Beispiel aus dem 4. Minitest, Frage 9.

23) Seien $X_i \sim Bernoulli(p), i \in [n]$, dann ist $(\sum_{i=1}^{n} X_i) \sim Bin(n, p)$

Die n Bernoulli Variablen müssen unabhängig sein.

24) Seien $X, Y \sim Bin(32, 0.25)$ unabhängig. Dann ist Var$[X - Y] =$

$\text{Var}[X - Y] = \text{Var}[X] + \text{Var}[Y] = 2 \cdot (32 \cdot 0.25 \cdot (1 - 0.25)) = 12$

25) Der Coupon Collector hat die n Karten schneller gesammelt, falls

er die letzten $0.5n$ Karten geschenkt bekommt.

Erwartungswert, wenn man die ersten k Karten kriegt: $\mathbb{E}[X] = n \cdot H_{n-k}$

Erwartungswert, wenn man die letzten k Karten kriegt: $\mathbb{E}[X] = n \cdot (H_n - H_k)$

Für $k = 0.5n$ ist der zweite Term kleiner.

26) Sei X eine Zufallsvariable mit $\mathbb{E}[X] = 10$. Dann gilt $\Pr[X \geq 20] \leq 0.5$

Vorischt, Markovs Ungleichung gilt nur für nicht-negative X. Man könnte z.B. X so definieren, sodass $\Pr[X = 100] = 0.25$ und $\Pr[X = -20] = 0.75$, dann gilt die Aussage nicht.

27) Wie viele englische Könige sind auf der Toilette gestorben?

https://en.wikipedia.org/wiki/List_of_people_who_died_on_the_toilet

28) Sei $X \sim Bin(32, 0.25)$. Was gibt uns eine stärkere obere Schranke für $\Pr[X \geq 12]$**?**

Mit Markov kriegt man $\frac{2}{3}$, mit Chebyshev $\Pr[X \geq 12] = \Pr[X - 8 \geq 4] \leq \Pr[|X - \mathbb{E}[X]| \geq 4] \leq \frac{3}{8}$

29) Welche Algorithmen haben eine randomisierte Laufzeit?

Las Vegas. Nie falsch, aber Laufzeit Random.

Monte Carlo sind manchmal falsch, aber haben eine fixe Laufzeit.

30) Was passiert mit der Varianz unserer Approximation, wenn wir beim Target-Shooting mehr zufällige Punkte wählen?

$\mathrm{Var}[Y] = \frac{1}{N}\left(\frac{|S|}{|U|} - \left(\frac{|S|}{|U|}\right)^2\right)$, somit ist die Varianz streng monoton fallend in $N =$ Anzahl der ausgewählten Punkte.

31) Wir wenden den Miller-Rabin Primzahltests auf eine natürliche Zahl n an. Dann gilt sicher:

Fall der Test "nicht prim" ausgibt, so hat er ein Zeuge dafür gefunden, dass n definitiv nicht prim ist. Ansonsten gibt es "prim" aus, weil er kein Zeuge gefunden hat und somit annimmt, dass die Zahl prim ist. Für eine Primzahl wird es natürlich nie ein Zeuge geben, somit wird der Algo immer "prim" asugeben.

32) Es gibt unbegrenzt viele Rosen von 4 Farben. Wir wollen 2 für einen Blumenstrauss wählen. Wie viele Möglichk. gibt es?

Wir wählen die Rosen ungeordnet und mit Zurücklegen, also haben wir $\binom{4+2-1}{2} = \frac{5!}{2! \cdot (5-2)!} = 10$ Möglichkeiten.

Detaillierte Begründung zur Herleitung der Formel (mit Strichen und Punkten): Wir machen 2 Punkte für die 2 Rosen unseres Blumenstrausses. Nun wollen wir die 2 Punkte auf unsere 4 Farben verteilen (beliebig). Dazu können wir 3 Striche platzieren, links, zwischen, oder rechts von den beiden Punkten (Farbe1 | Farbe2 | Farbe 3 | Farbe 4, als Beispiel: |*||* heisst 1 Rose von Farbe 2, 1 Rose von Farbe 4). Um die Anzahl zu berechnen, müssen wir also 2 aus 5 auswählen, was der obigen Formel entspricht: $\binom{5}{2} = 10$.

33) Sei $P \subset \mathbb{R}^2$ beliebig. Dann liegen höchstens 3 Punkte aus P auf $C(P)$.

Es können natürlich viel mehr Punkte drauf liegen. Was wichtig ist, ist dass es nie mehr als 3 essentielle Punkte gibt.

34) Wie schnell findet Jarvis Wrap die nächste Ecke einer konvexen Hülle von n Punkten im Gegenuhrzeigersinn?

$\mathcal{O}(n)$, weil man für $n-2$ Punkte den rechts-Test machen muss.

35) Wir haben durch Boostrapping die Laufzeit des Cut Algorithmus von $\mathcal{O}(n^4)$ auf $\mathcal{O}(n^3)$ verbessert, aber es geht noch besser.

Beim Bootsrapping haben wir unseren $\mathcal{O}(n^4)$ Algo auf die letzten t Knoten angewendet und somit die Laufzeit verbessert. Wenn man nun den neuen $\mathcal{O}(n^3)$ Algo auf die letzten t Knoten anwendet, kriegt man noch eine bessere Laufzeit.

You made it through the whole Kahoot!! Big congratulations, here is a cute cat that wishes you good luck in your exams: