



Material zum Unterricht

Dr. Kai-Friederike Oelbermann

Alle Folien, Hausaufgaben, Links, Termine etc. finden Sie unter:

www.kai-friederike.de

Benutzername: mathe

Passwort: isttoll

GS Informatik (Wintersem

Unterricht

Tag	Uhrzeit	Raum
Montags	12h00-13h30	303
Donnerstags	12h00-13h30	303

Lehrplan

[Hier](#) finden Sie den Lehrplan.

Material

Persönliches:

E-Mail: Kai-Friederike.Oelbermann@hs-anhalt.de

CV:

- seit 1. juni 2017 arbeite ich im [Präsidialbüro](#) sowie am [Landesstudienkolleg](#) der [Hochschule Anhalt](#).
- Mai 2015 bis Mai 2017: Wissenschaftliche Mitarbeiterin am [Institut für Mathematische Optimierung](#) an der Otto-von-Guericke Universität Magdeburg.
- April 2014 bis Mai 2015: Lehrbeauftragte an der Hochschule Magdeburg-Stendal
- Mai bis Oktober 2013: Referentin bei der Studienstiftung des deutschen Volkes
- Mai 2009 bis Mai 2013 wissenschaftliche Mitarbeiterin an der Universität Augsburg.



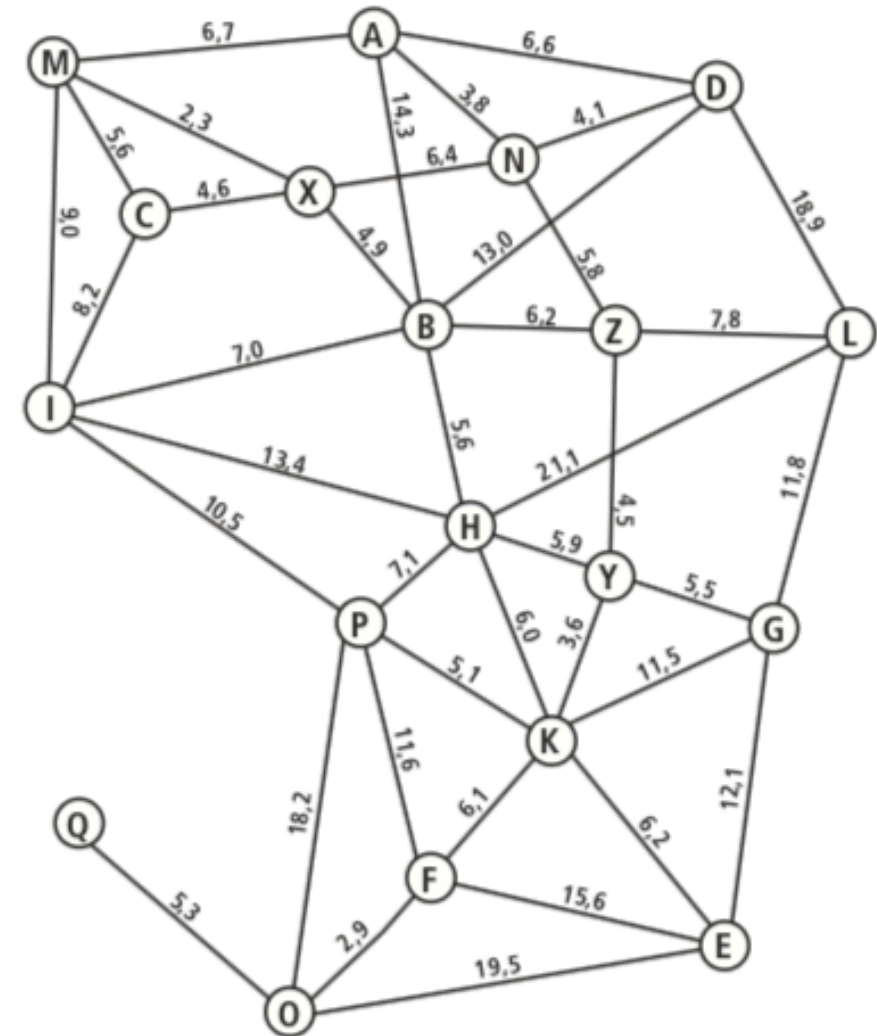
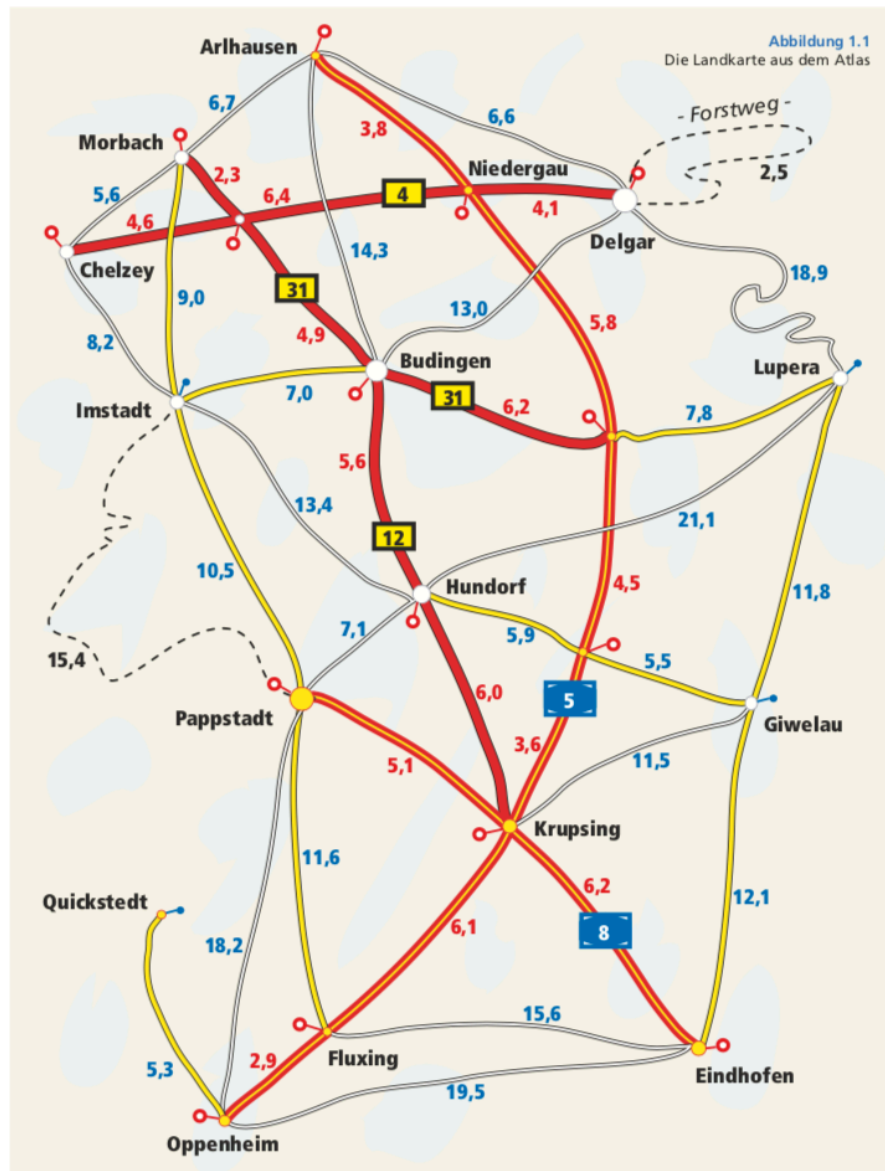
Lehre:

Wintersemester 2018/19 (Landesstudienkolleg, Hochschule Anhalt, Standort Köthen)

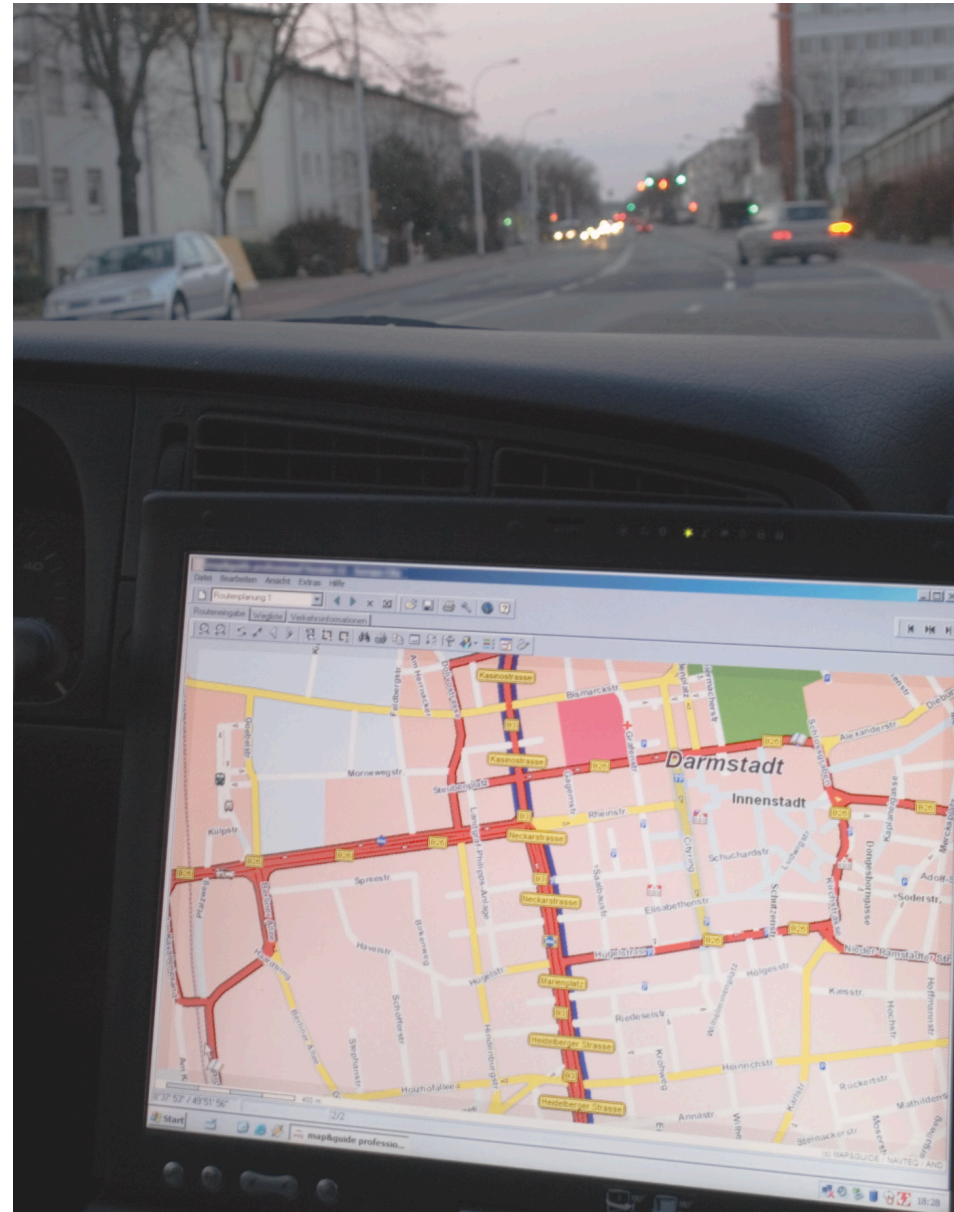
- [TH-2](#) (Mathematik)
- [GS](#) (Informatik)

**„Informatik ist der Schlüssel
zum Verständnis der
Informationstechnik und
damit der Schlüssel zum
Verstehen unserer modernen
Umwelt!“**

Dijkstra und die Ameisen



Aufgabe: Versuchen Sie den kürzesten Weg von Imstadt nach Oppenheim zu finden (Lösung: 24,6 km).



Dijkstra und die Ameisen

Definition: Methode der Abstraktion

In zur Verfügung stehender Information stecken sowohl relevante als auch un- wesentliche Anteile. Durch Abstraktion reduzieren Sie die Information auf das für die aktuelle Problemlösung Wesentliche: Dadurch können Sie sich besser auf Ihre Aufgabe konzentrieren.

Definition: Methode der Gleichformung

Versuchen Sie, die verschiedenen Facetten eines Problems auf die gleichen Grundelemente zurückzuführen. Dadurch wird einerseits das Problem über- sichtlicher und andererseits benötigt man weniger Lösungsansätze: Für gleichförmige Teilprobleme kann der gleiche Lösungsansatz verwendet werden.

Linie 1



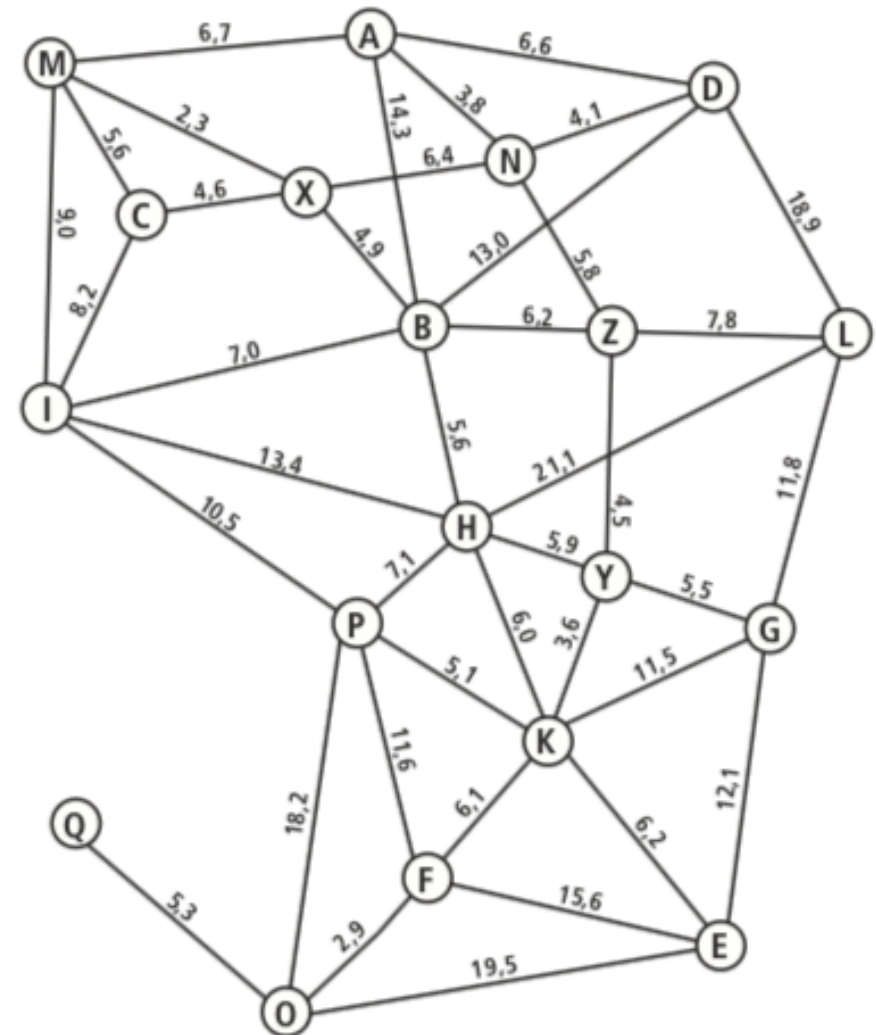
Abstraktion begegnet uns ständig! Wegweiser, Straßenschilder, Fahrpläne, Infobroschüren und viele andere Dinge des Alltags sind so aufbereitet, dass die nötigen Informationen möglichst offen und gut erkennbar dargestellt sind.



Auch **abstrakte Malerei** ist die Konzentration auf das Wesentliche. Der Künstler stellt die Aspekte in den Mittelpunkt, die ihn bewegen, zum Beispiel ein Gefühl oder ein Ereignis. Details wie die realistische Darstellung treten dadurch in den Hintergrund. Hier abgebildet ist das „Blaue Pferd“ von Franz Marc (1911).

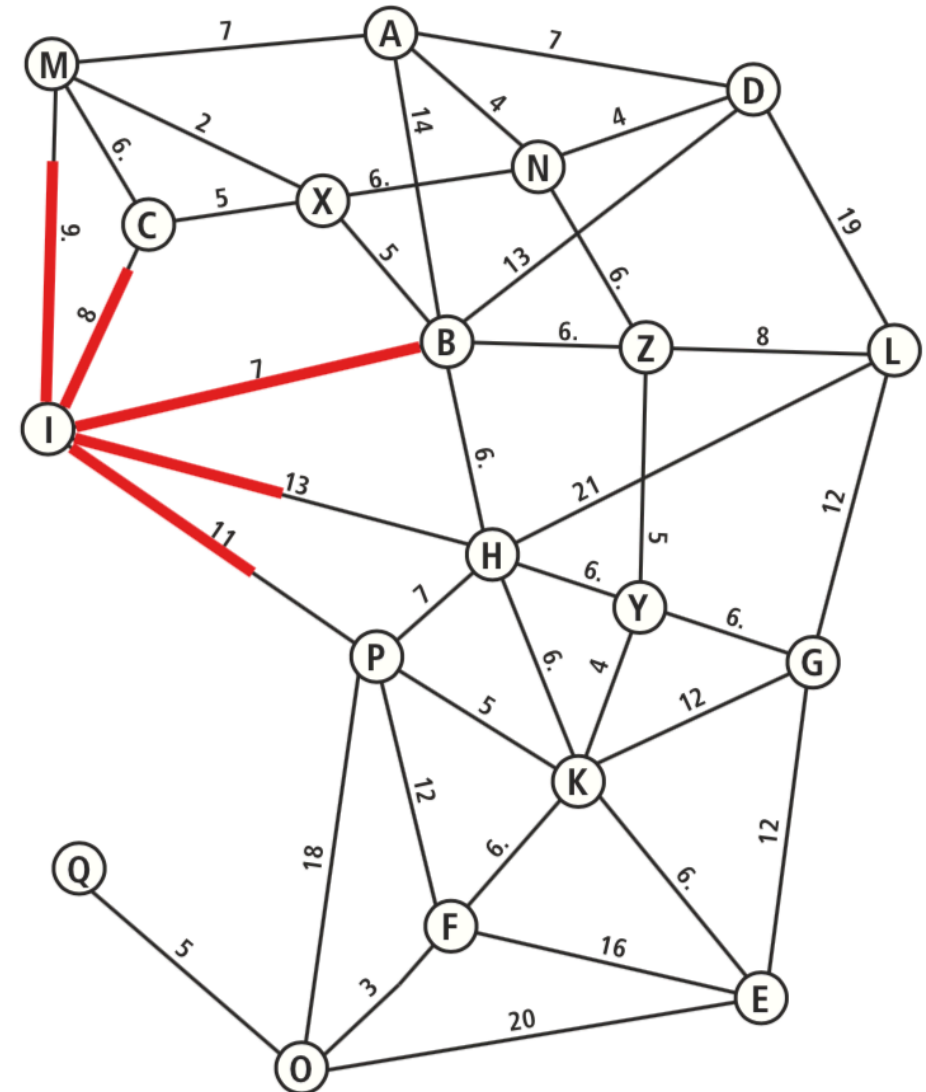
Lösungsmethoden

- Alle Strecken ausprobieren = **Brute-Force-Algorithmus**
- Problem: Dauert bei großen Problemen sehr lange.
- Daher: Einen bessere/effizientere Methode finden!



Dijkstra und die Ameisen

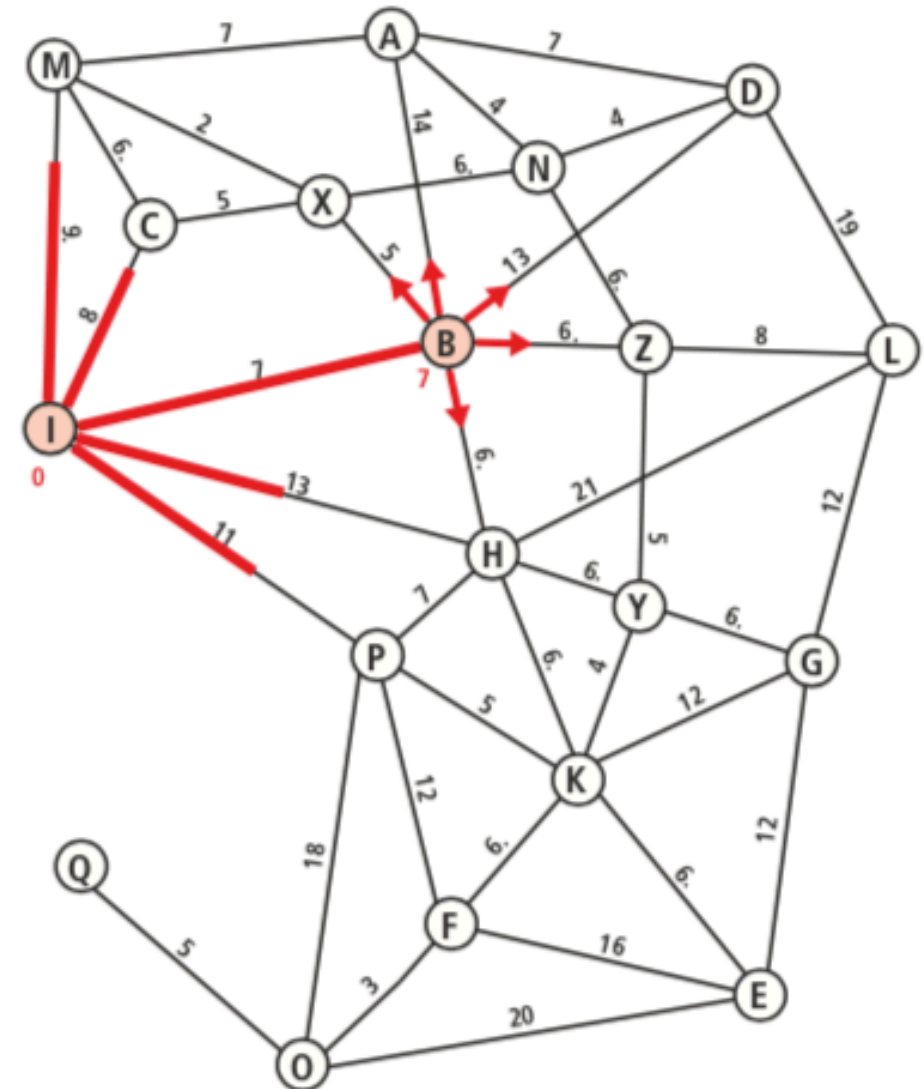
Mit dem Finger auf der Landkarte verfolgen wir den Weg der Ameisen. Diese Abbildung zeigt in Rot ihren Fortschritt nach 7 Minuten.



Dijkstra und die Ameisen

Mit dem Finger auf der Landkarte verfolgen wir den Weg der Ameisen. Diese Abbildung zeigt in Rot ihren Fortschritt nach 7 Minuten.

Die Ameisen, die bisher nirgendwo angekommen sind, setzen ihren Weg einfach fort. Die Ameisen bei B teilen sich erneut auf: Wieder sind fünf Wege möglich.

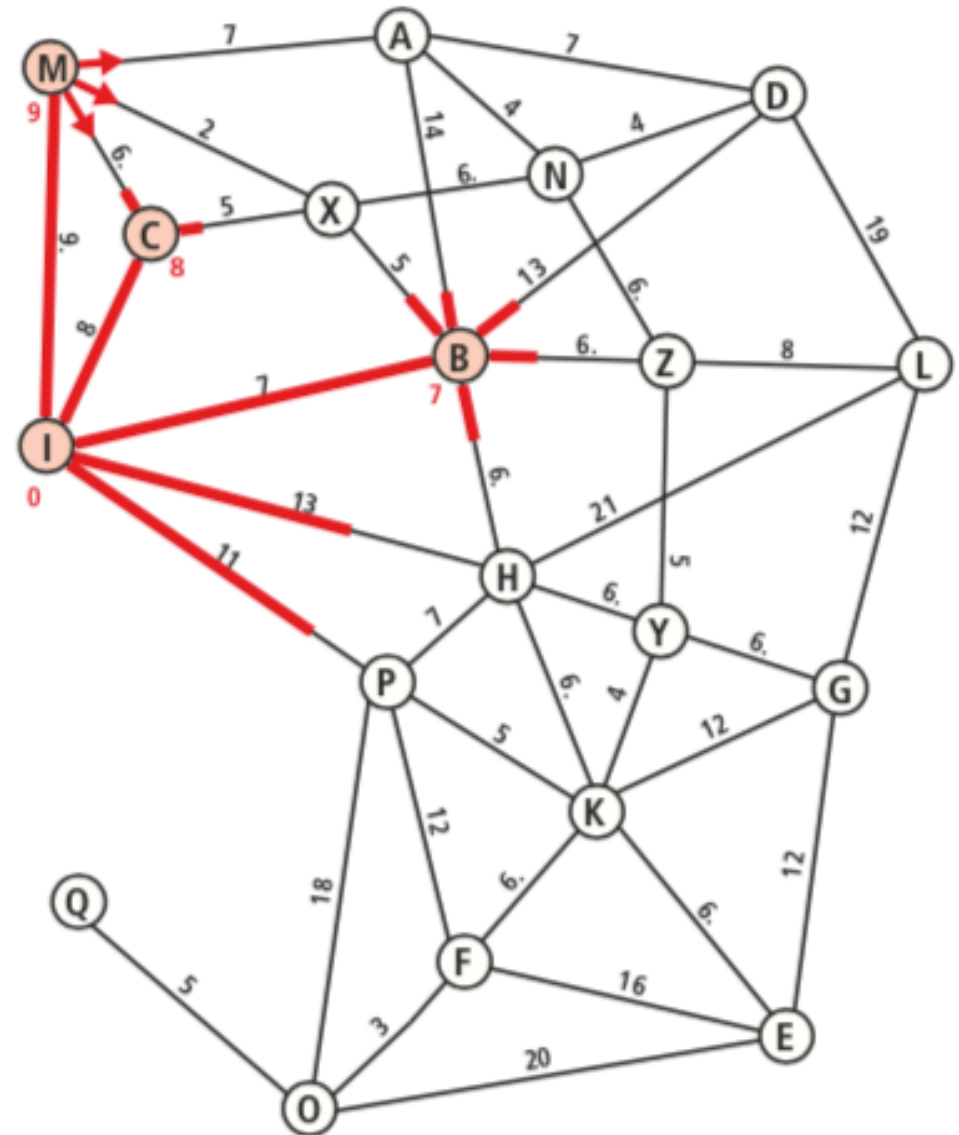


Dijkstra und die Ameisen

Mit dem Finger auf der Landkarte verfolgen wir den Weg der Ameisen. Diese Abbildung zeigt in Rot ihren Fortschritt nach 7 Minuten.

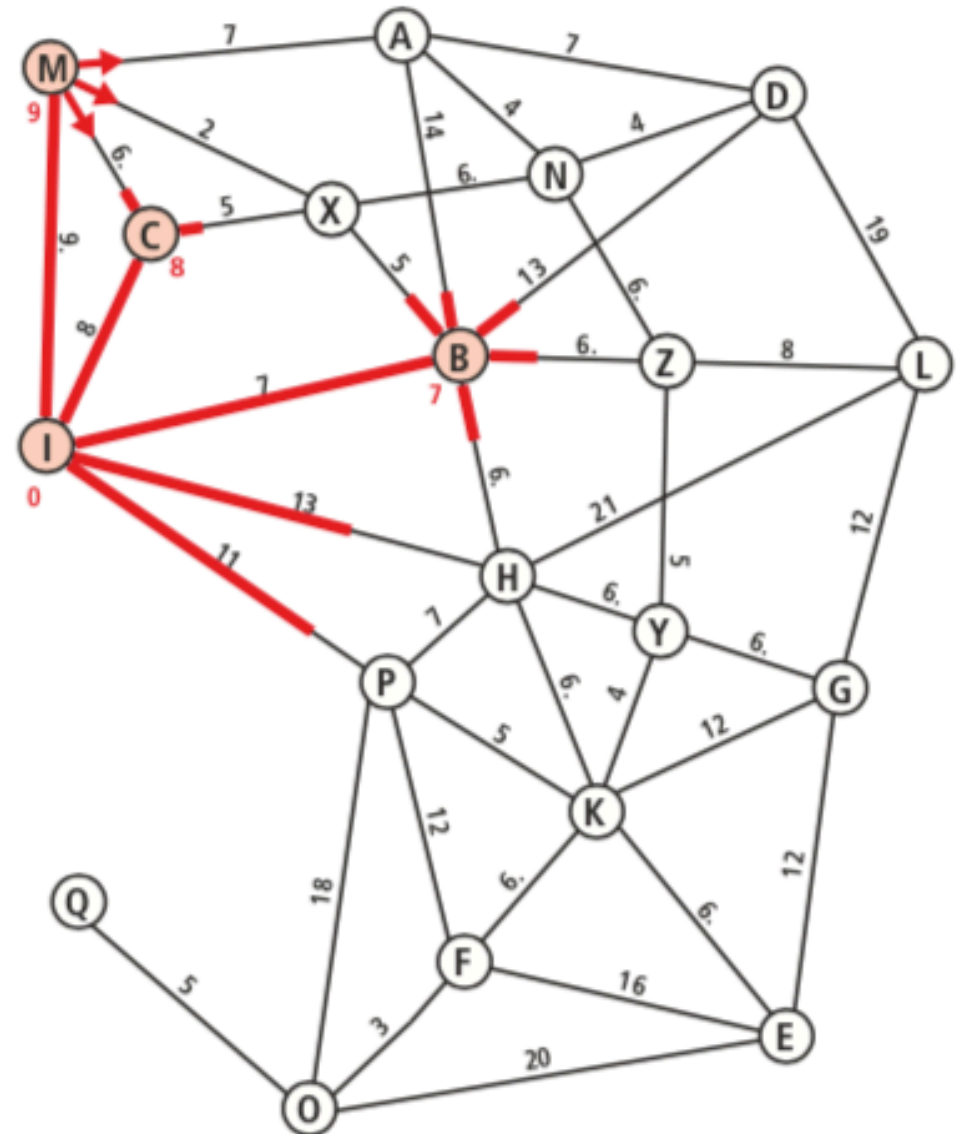
Die Ameisen, die bisher nirgendwo angekommen sind, setzen ihren Weg einfach fort. Die Ameisen bei B teilen sich erneut auf: Wieder sind fünf Wege möglich.

Am Ende der neunten Minute kommt dann auch der Trupp bei M als Erster an. Auch dieser Weg wird vermerkt



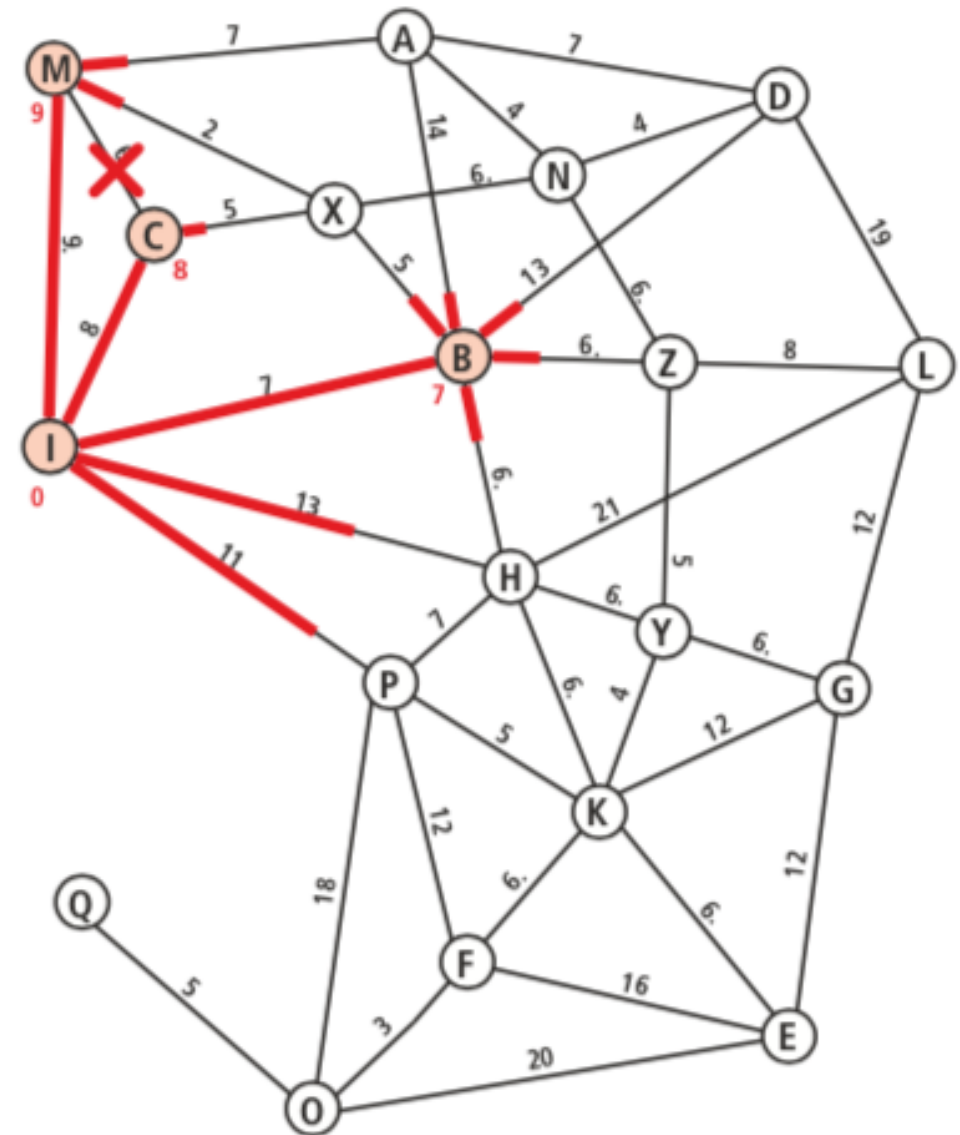
Dijkstra und die Ameisen

Vielleicht haben Sie bemerkt, dass nun Ameisentrupps sowohl von M als auch von C ausgehend unterwegs sind – zwischen beiden Städten auf Kollisionskurs. Welche Informationen können sie austauschen?



Dijkstra und die Ameisen

Der Trupp von C weiß, dass dieses Ziel bereits erreicht ist, die kürzeste Strecke dorthin also feststeht. Der Trupp, der von M kommt, kann das Gleiche von seinem Ausgangspunkt berichten. Es ist also sinnlos, weiterzumarschieren. Die Strecke wird als „unbrauchbar“ markiert. Diese Abbildung zeigt das gleich mit einem dicken roten Kreuz.

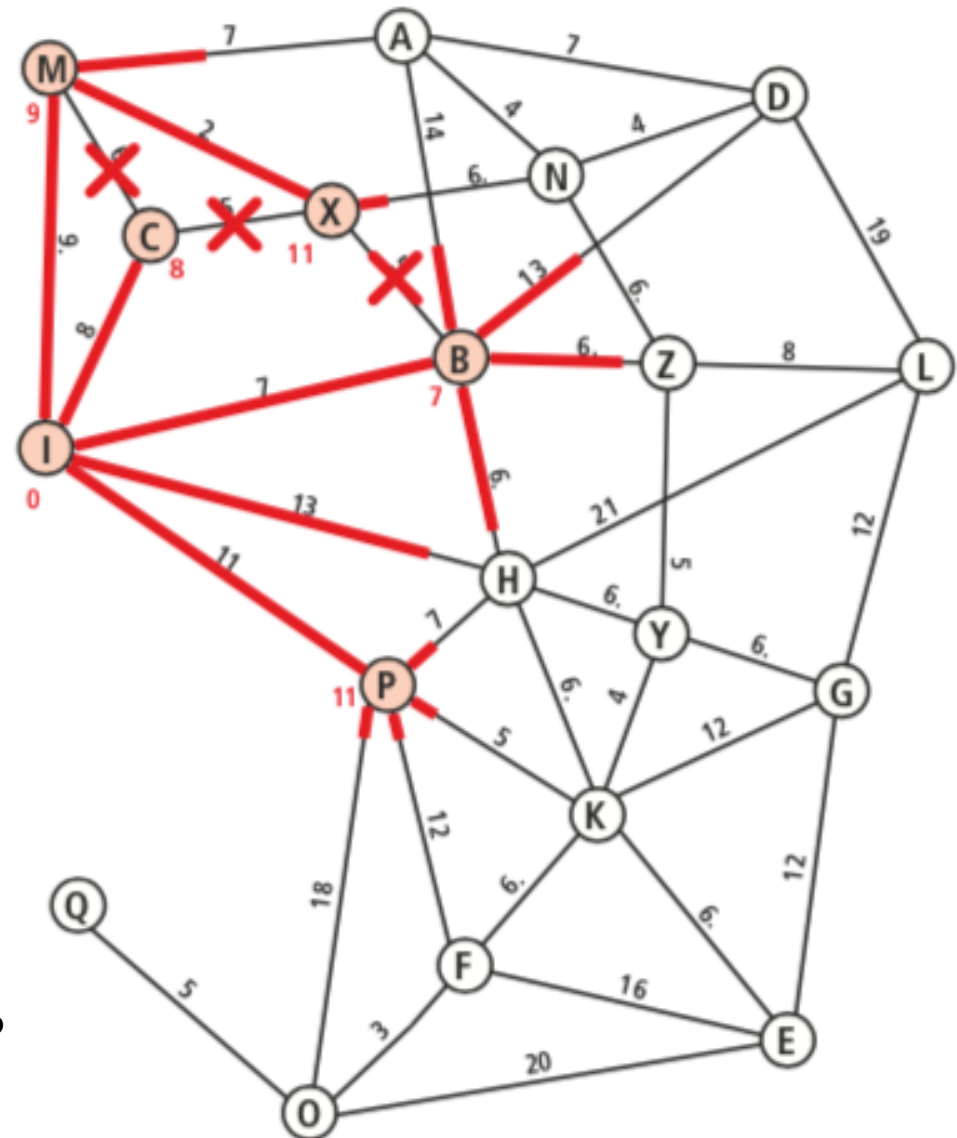


Dijkstra und die Ameisen

Der Trupp von C weiß, dass dieses Ziel bereits erreicht ist, die kürzeste Strecke dorthin also feststeht. Der Trupp, der von M kommt, kann das Gleiche von seinem Ausgangspunkt berichten. Es ist also sinnlos, weiterzumarschieren. Die Strecke wird als „unbrauchbar“ markiert. Diese Abbildung zeigt das gleich mit einem dicken roten Kreuz.

Als Nächstes kommen zwei Erkundungstrupps gleichzeitig an: In der 11. Minute erreichen die Ameisen P und X. P erreichen sie auf direktem Wege von I aus. Bei X kommt der Trupp an, der über M unterwegs ist (9 km bis M plus 2 bis X).

Wieder teilen sich die Ameisen auf. Von X gibt es nur noch einen erfolgversprechenden Weg, bei dem anderen treffen sie recht schnell auf Kameraden und geben die Strecke auf. Die von P ausgehenden Touren sind alle noch offen. Diese Abbildung zeigt den aktuellen Stand.

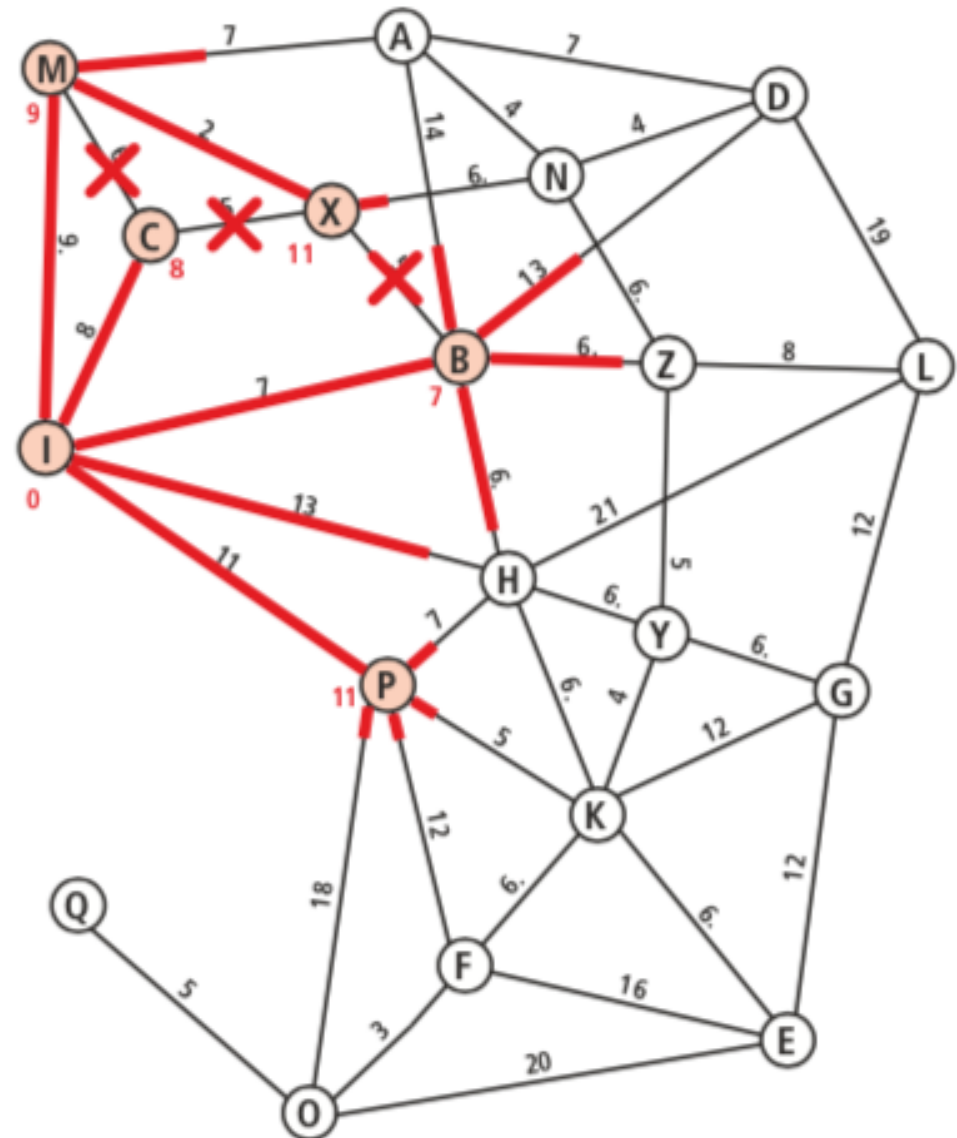


Dijkstra und die Ameisen

Warum ist das Ameisen-Prinzip für einen Informatiker interessant?

- Es führt in absehbarer Zeit zum Ziel. Da die Ameisen ständig in Bewegung sind und keine Strecke doppelt gehen, müssen sie recht bald alles erkundet haben (maximal nach der Zeit, die dem kürzesten Weg zur am weitesten entfernten Stadt entspricht).
- Es werden immer wieder die gleichen, sehr einfachen Anweisungen benutzt, um die Ameisen zu steuern:
 - ① Teile den Trupp auf und folge allen Routen.
 - ② Wenn ein Ort erreicht wird: kürzeste Strecke dorthin gefunden, weiter bei ①.
 - ③ Wenn man einem anderen Trupp begegnet: Strecke verwerfen, Ende.

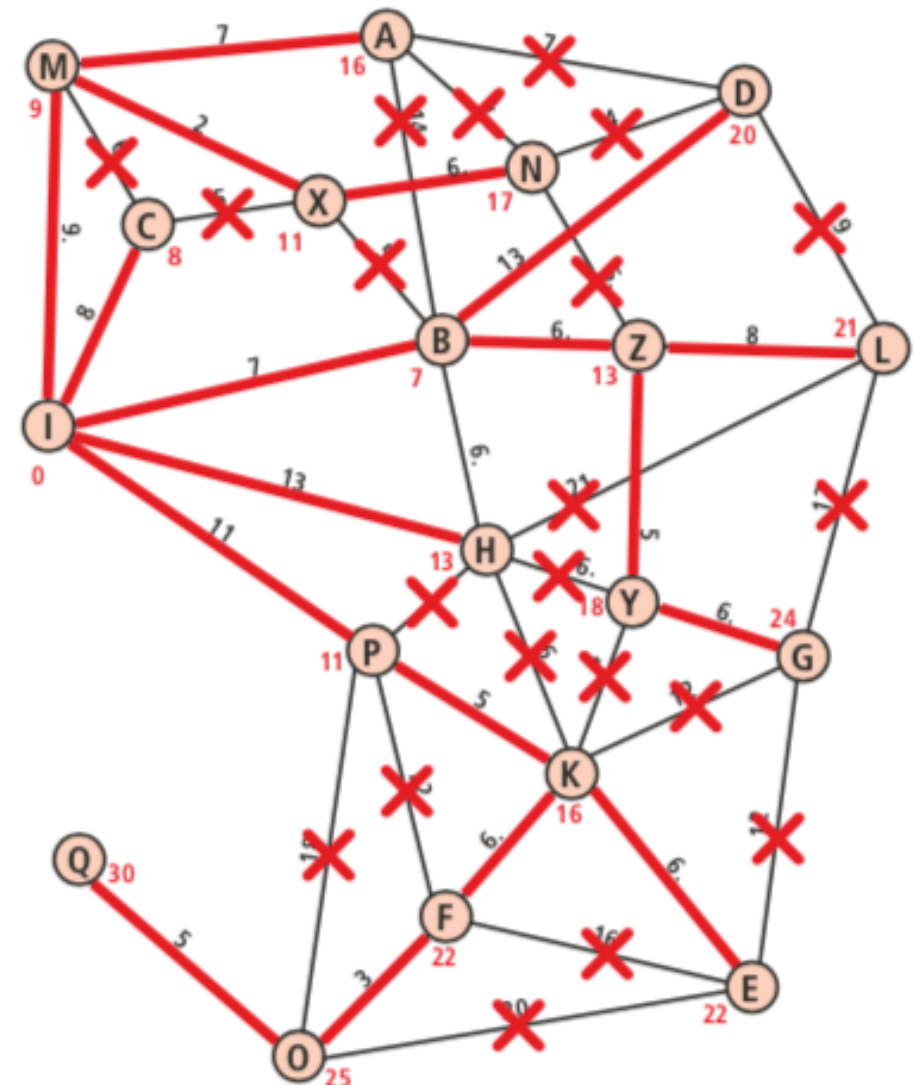
Aufgabe: Machen Sie weiter! Gesucht ist die kürzeste Strecke von I nach O.



Dijkstra und die Ameisen

Warum ist das Ameisen-Prinzip für einen Informatiker interessant?

- Es führt in absehbarer Zeit zum Ziel. Da die Ameisen ständig in Bewegung sind und keine Strecke doppelt gehen, müssen sie recht bald alles erkundet haben (maximal nach der Zeit, die dem kürzesten Weg zur am weitesten entfernten Stadt entspricht).
- Es werden immer wieder die gleichen, sehr einfachen Anweisungen benutzt, um die Ameisen zu steuern:
 - ❶ Teile den Trupp auf und folge allen Routen.
 - ❷ Wenn ein Ort erreicht wird: kürzeste Strecke dorthin gefunden, weiter bei ❶.
 - ❸ Wenn man einem anderen Trupp begegnet: Strecke verwerfen, Ende.



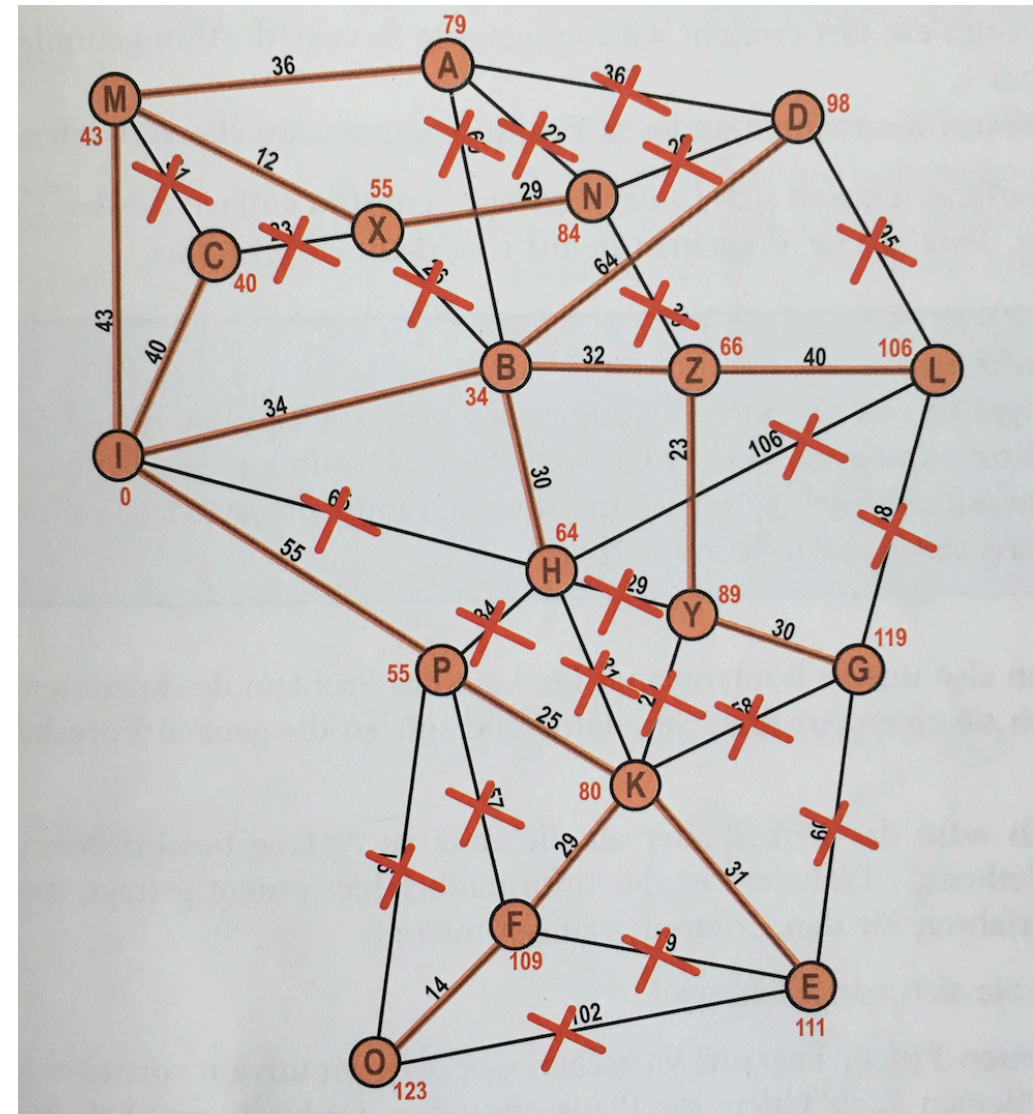
Aufgabe: Machen Sie weiter!

Lösung: kürzeste Wege, Buch (4. Auflage)

Dijkstra und die Ameisen

Warum ist das Ameisen-Prinzip für einen Informatiker interessant?

- Es führt in absehbarer Zeit zum Ziel. Da die Ameisen ständig in Bewegung sind und keine Strecke doppelt gehen, müssen sie recht bald alles erkundet haben (maximal nach der Zeit, die dem kürzesten Weg zur am weitesten entfernten Stadt entspricht).
- Es werden immer wieder die gleichen, sehr einfachen Anweisungen benutzt, um die Ameisen zu steuern:
 - ❶ Teile den Trupp auf und folge allen Routen.
 - ❷ Wenn ein Ort erreicht wird: kürzeste Strecke dorthin gefunden, weiter bei ❶.
 - ❸ Wenn man einem anderen Trupp begegnet: Strecke verwerfen, Ende.



Aufgabe: Machen Sie weiter!

Lösung: kürzeste Wege auf der Kopiervorlage

Dijkstra und die Ameisen

Definition: Algorithmus

Ein Algorithmus ist eine Handlungsvorschrift zur Lösung eines Problems bzw. einer Kategorie von Problemen. Diese Handlungsvorschriften lassen sich im Allgemeinen in ein Computerprogramm umsetzen. Hierfür müssen sie hinreichend genau formuliert sein.

Aufgabe: Versuchen Sie zunächst einmal selbst, ein solches Verfahren herauszufinden. Vielleicht kommen Sie ja sogar auf eine bessere Methode, die dann nach Ihnen benannt wird.

Algorithmus

Wussten Sie, dass die Herkunft dieses Begriffes nie richtig geklärt wurde? Entweder kommt er vom griechischen „arithmos“, was Zahl bedeutet, oder er ist der veränderte Name des persisch-arabischen Mathematikers „Al-Charismi“, der – englisch ausgesprochen – in etwa wie „algorithm“ klingt. Erste Algorithmen wurden übrigens bereits lange vor der Zeitenwende hauptsächlich von griechischen Mathematikern entwickelt: Etwa im 4. Jahrhundert v. Chr. das euklidische Verfahren zur Bestimmung des größten gemeinsamen Teilers zweier Zahlen oder im 3. Jahrhundert v. Chr. das Sieb des Eratosthenes zur Ermittlung von Primzahlen. Beide Algorithmen werden heute noch in Computern eingesetzt!

Aufgabe 1:

1. Suche Smartphone oder PC mit Internetanschluss
2. Gehe zu Seite: www.inf.hs-anhalt.de/ActivateAccount
3. Geben Sie Benutzernamen (z.B. Oelbermann_k) und Initialpasswort (8 Zeichen, z.B. DY8=A3B6) ein.

Wenn Sie das gemacht haben, können Sie sich am Donnerstag im PC Pool einloggen! Juhu!!

Aufgabe 2:

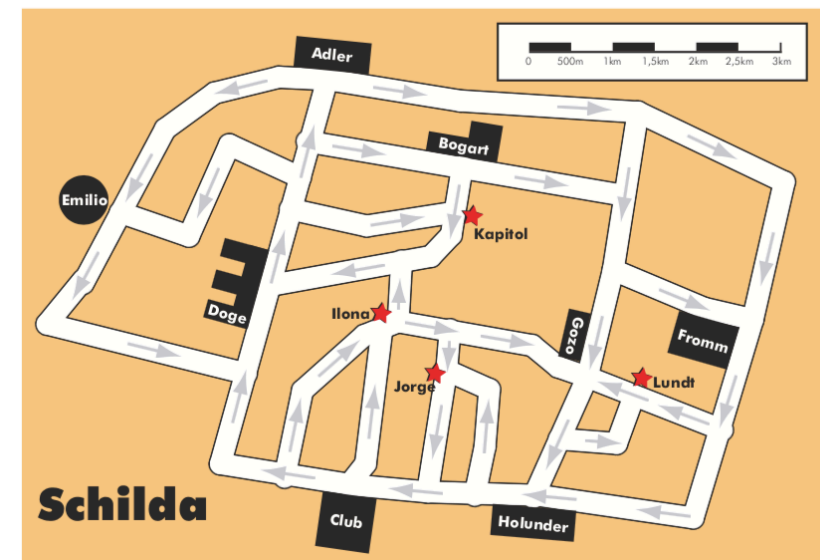
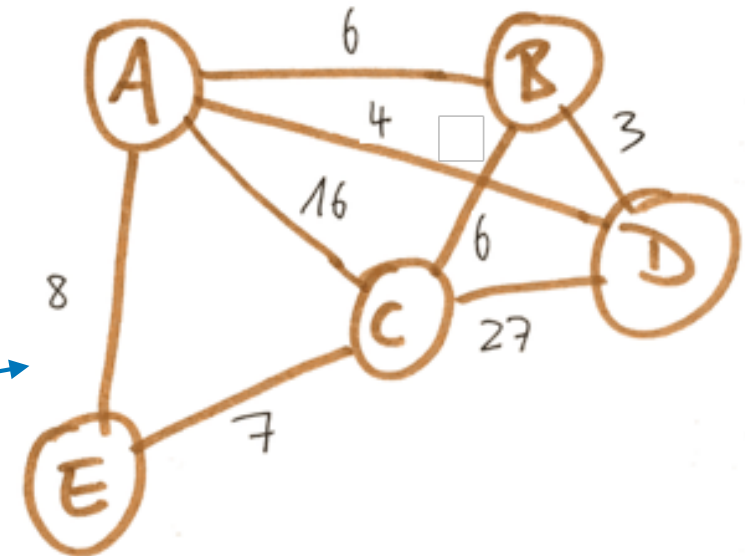
Suchen Sie hier, ähnlich wie zuvor, die kürzesten Wege ausgehende von A. Schreiben Sie jeweils an die Städte, wie lang die kürzesten Wege sind.

Hausaufgabe 3:

Suchen Sie in dem großen Graph, ähnlich wie zuvor, die kürzesten Wege ausgehend von der Stadt Delgar!

Aufgabe 4:

Abstrahieren Sie den Stadtplan von Schilda und erstellen Sie drei Tabellen mit den Entfernungen der Hotels Adler und Gozo sowie der Pension Kapitol zu allen anderen Gästehäusern.



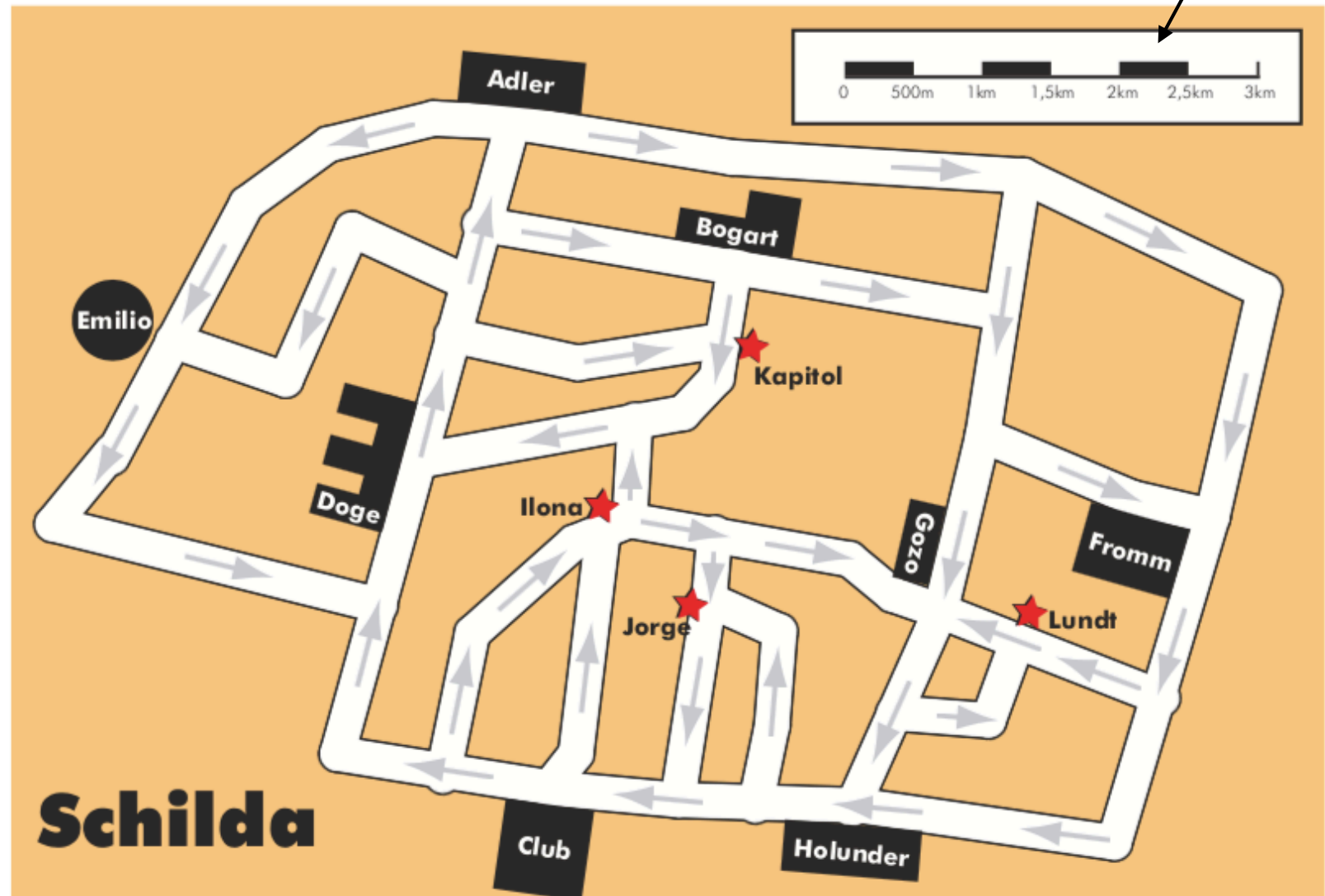


Aufgabe 4:

Abstrahieren (vereinfachen) Sie den Stadtplan von Schilda und erstellen Sie **drei Tabellen mit den Entfernungen der Hotels Adler und Gozo sowie der Pension Kapitel** zu allen anderen Gästehäusern.

Der Unterschied zum letzten Problem sind die Einbahnstraßen.

Maßstab



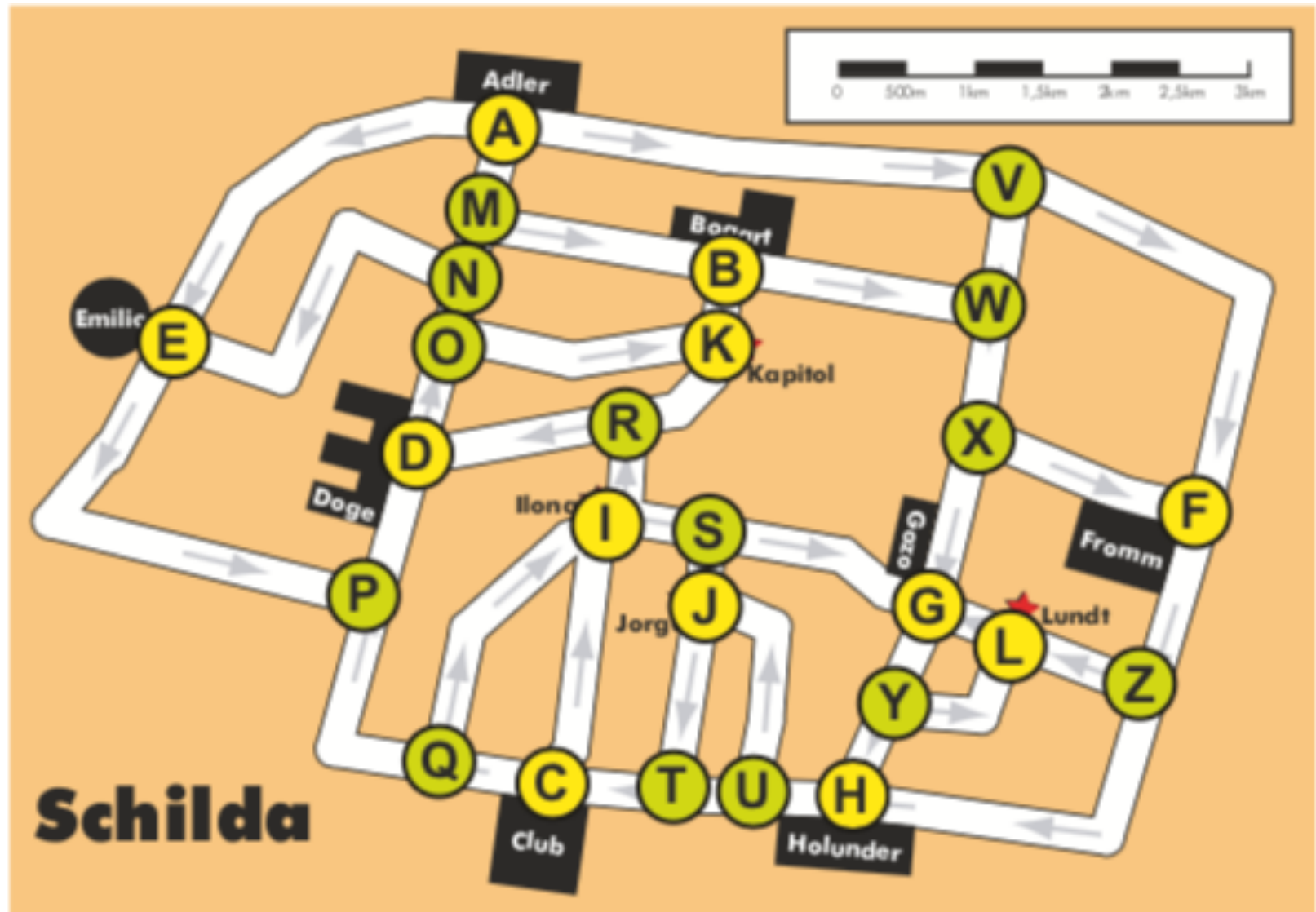
Aufgabe 4:

Abstrahieren Sie den Stadtplan von Schilda und erstellen Sie drei Tabellen mit den Entfernungen der Hotels Adler und Gozo sowie der Pension Kapitel zu allen anderen Gästehäusern.

Hinweise:

1. Haben Sie die Biegungen zwischen E und P oder zwischen V und F markiert? Das ist kein Fehler, jedoch nicht notwendig, denn ein Auto kann sowieso nur dem Straßenverlauf folgen.

2. Der Unterschied zum letzten Problem sind die Einbahnstraßen.



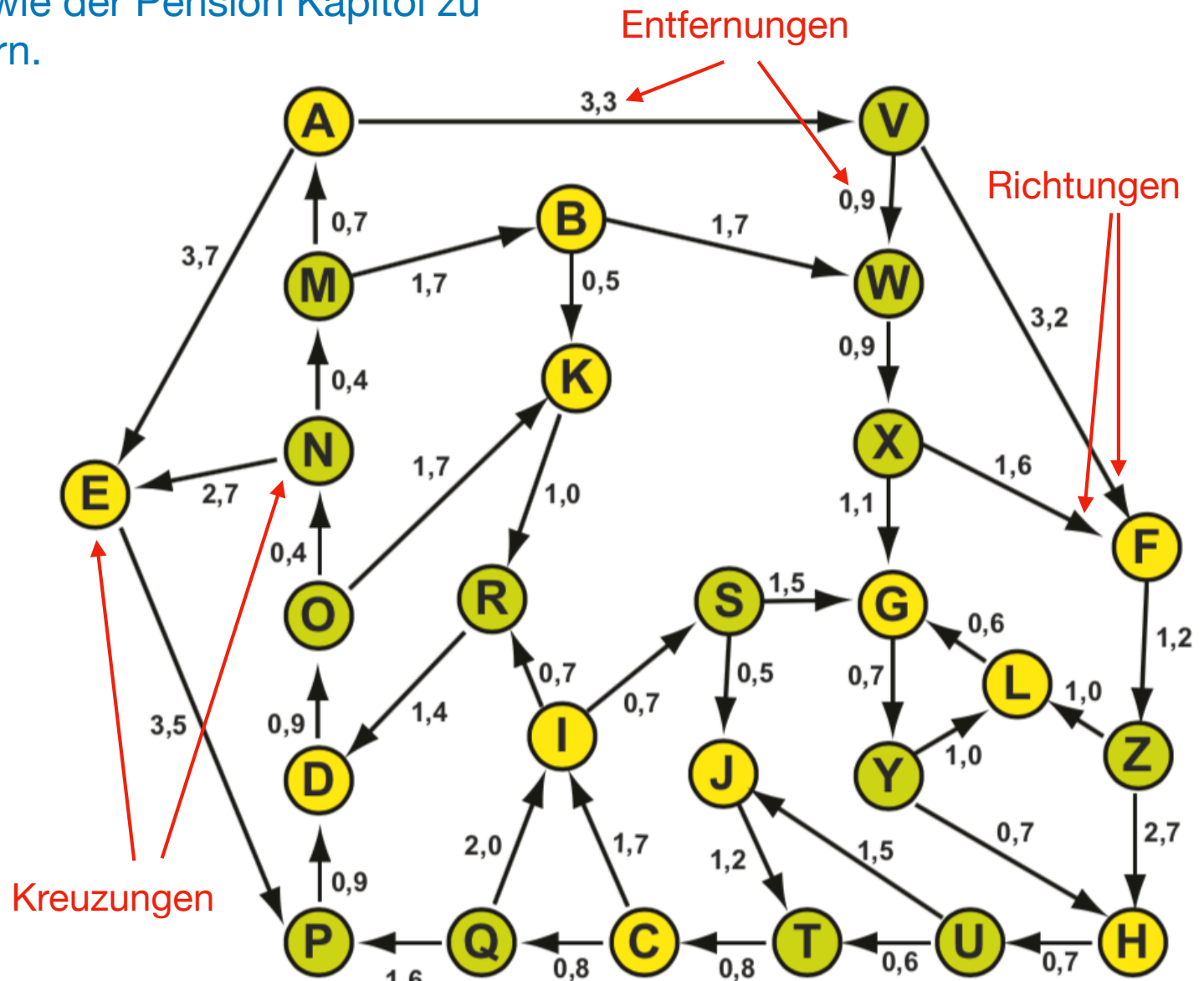
Aufgabe 4:

Abstrahieren Sie den Stadtplan von Schilda und erstellen Sie drei Tabellen mit den Entfernungen der Hotels Adler und Gozo sowie der Pension Kapitel zu allen anderen Gästehäusern.

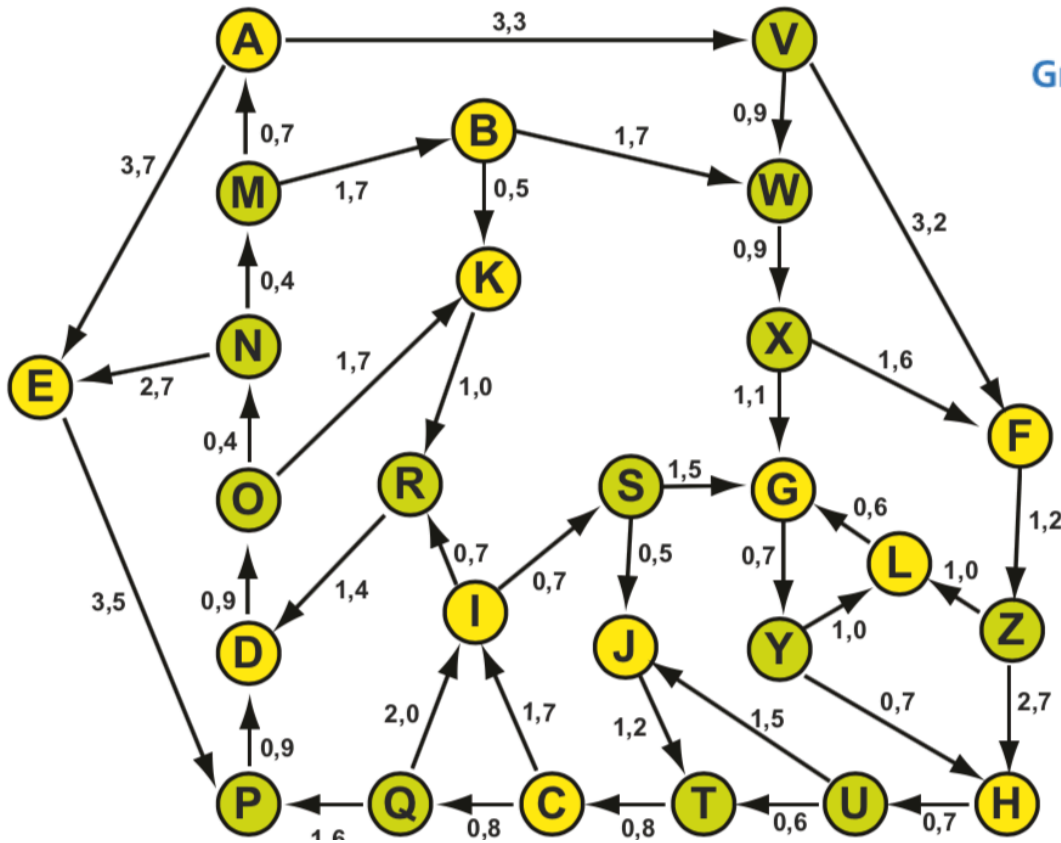
Kürzeste Entfernungen vom Hotel Adler. Sie ergeben folgende Tabelle:

Hotel	Entfernung
Adler	0,0
Bogart	11,5
Club	9,7
Doge	8,1
Emilio	3,7
Fromm	6,5
Gozo	6,2
Holunder	7,6
Ilona	11,4
Jorge	9,8
Kapitol	10,7
Lundt	7,9

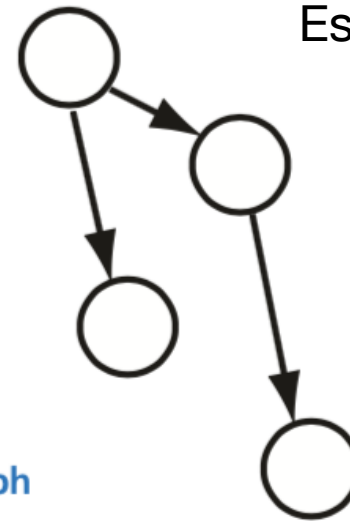
LÖSUNG



- Die besprochenen Probleme gehören in das Umfeld der sogenannten **Graphenalgorithmien**.
- Ein **Graph** besteht aus einer Menge von **Knoten** und einer Menge von **Kanten**, die zwischen den Knoten verlaufen.



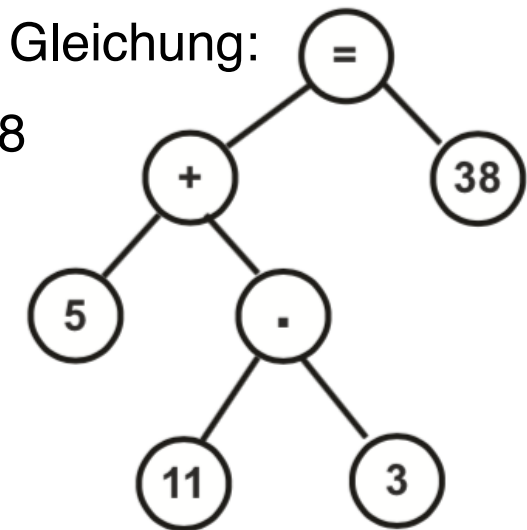
Graph



Es gibt **gerichtete** Graphen.

Mathematische Gleichung:

$$5 + (11 \cdot 3) = 38$$



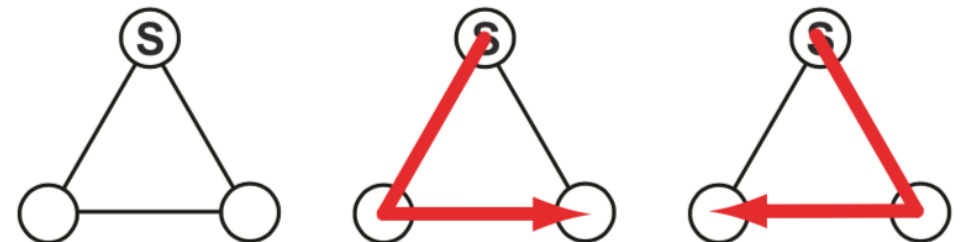
Mathematische Gleichung

Es gibt **ungerichtete** Graphen.

Vollständige Graphen

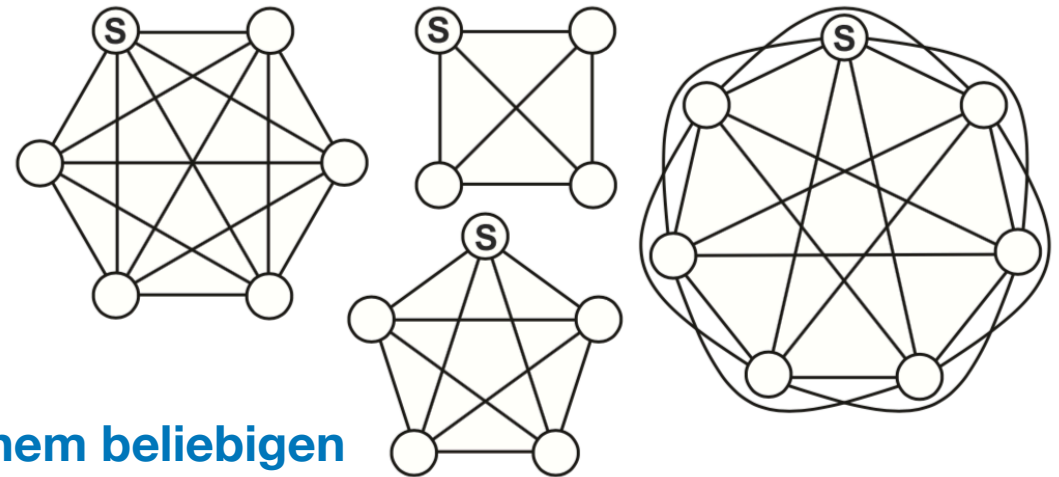
- Brute-Force-Algorithmus: „Bestimme alle möglichen Wege vom Start zum Ziel und suche davon den kürzesten.“
- Im schlechtesten Fall müssen also von einem Graphen **alle von einem Punkt ausgehenden Wege** bestimmt werden. Außerdem sind im schlechtesten Fall alle Knoten mit allen anderen Knoten verbunden (der Fachausdruck dafür ist „**vollständiger Graph**“).

Es gibt je zwei Wege um von einem Knoten zu einem anderen zu gelangen.



Vollständige Graphen

- Brute-Force-Algorithmus: „Bestimme alle möglichen Wege vom Start zum Ziel und suche davon den kürzesten.“
- Im schlechtesten Fall müssen also von einem Graphen **alle von einem Punkt ausgehenden Wege** bestimmt werden. Außerdem sind im schlechtesten Fall alle Knoten mit allen anderen Knoten verbunden (der Fachausdruck dafür ist „**vollständiger Graph**“).



Aufgabe: Wie viele Wege gibt es von einem beliebigen Startknoten zu einem beliebigen Zielknoten in einem vollständigen Graphen mit 4, 5, 6 bzw. 7 Knoten?

Vollständige Graphen

LÖSUNG:

3 Knoten: 2 Wege

4 Knoten: 6 Wege

5 Knoten: 24 Wege

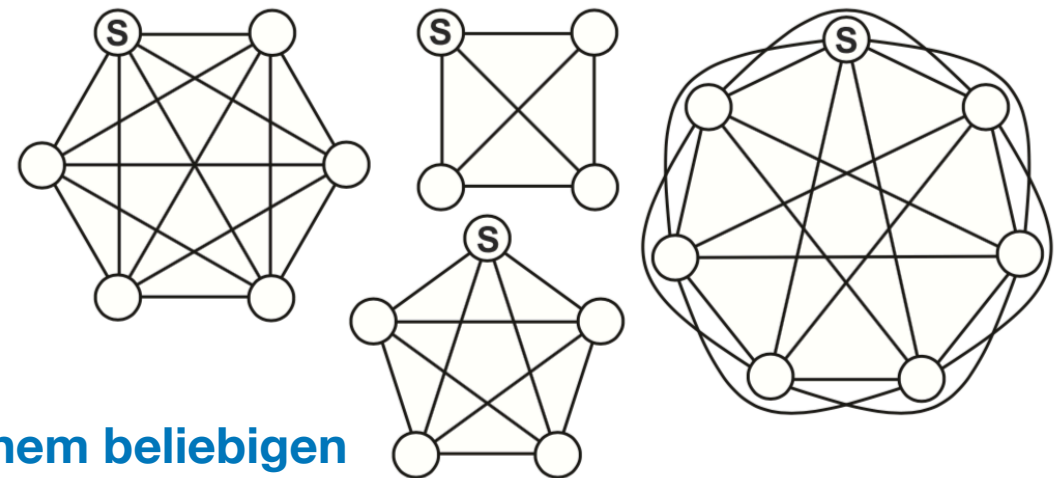
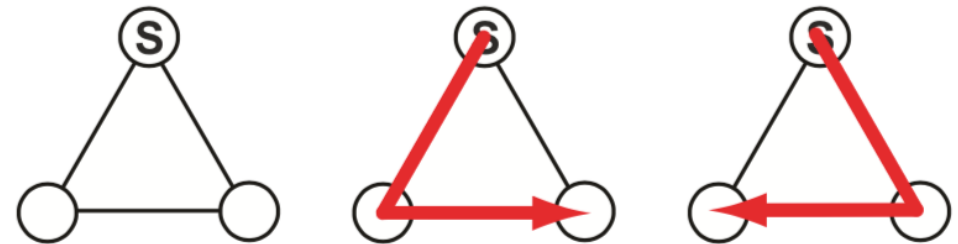
6 Knoten: 120 Wege

7 Knoten: 720 Wege

Allgemein gilt:

$$(n - 1)! = (n - 1) \cdot (n - 2) \cdot \dots \cdot 3 \cdot 2 \cdot 1$$

Es gibt je zwei Wege um von einem Knoten zu einem anderen zu gelangen.



Aufgabe: Wie viele Wege gibt es von einem beliebigen Startknoten zu einem beliebigen Zielknoten in einem vollständigen Graphen mit 4, 5, 6 bzw. 7 Knoten?





Aufgabe 1: Warm Up

1. Nehme ca. 60 Karten
2. Sortiere nach blauen Zahlen

187650	302782	503710	861867	212088	522107	38897	955099	96588	424261	747690	44948	581176	133986	467199	145423
aber	Ablage	absurd	Acker	Anbau	Anlage	Anteil	Arbeit	Aufruf	auftun	Ausland	Auslauf	Auster	Ausweg	Bank	Beere
276637	338210	425565	471452	544520	582278	260249	562184	444082	409121	559505	495493	163105	124890	448463	677562
488893	968277	154571	165413	9859	112092	553475	869328	885439	475340	598622	391932	539477	54218	369430	570546
Bein	bereit	Besitz	Beule	Birne	Boden	Brot	Busch	Dank	denn	Docht	Duft	Durst	ein	einmal	einzeln
551904	578884	708129	343721	310313	792094	497819	762015	328439	596164	578754	551017	303040	825829	431497	686943
495973	501884	737014	92280	251907	386655	903955	272409	902289	761556	245698	79474	666701	190206	525643	555035
Ente	Erde	erneut	Erz	falsch	Fell	Filter	Flug	frei	Fuhre	Garn	gefeit	Geist	genehm	Gesetz	gewagt
313203	545038	511851	416671	196973	348971	541849	456940	206299	511274	472166	541510	527355	501389	454812	533659
417230	770853	122091	220073	799190	316190	323682	791873	801774	645869	727378	242709	285853	468831	433707	822890
gleich	Graf	Gruppe	halb	Happen	Hefe	herb	Heu	hintere	hold	Idee	intern	jung	Karte	Kind	Klette
503130	493004	469414	502542	373529	286658	567975	411534	769872	587632	365762	500126	454879	504286	981432	656819



Aufgabe 1: Warm Up

1. Nehme ca. 60 Karten
2. Sortiere nach blauen Zahlen

187650	302782	503710	861867	212088	522107	38897	955099	96588	424261	747690	44948	581176	133986	467199	145423
aber	Ablage	absurd	Acker	Anbau	Anlage	Anteil	Arbeit	Aufruf	auftun	Ausland	Auslauf	Auster	Ausweg	Bank	Beere
276637	338210	425565	471452	544520	582278	260249	562184	444082	409121	559505	495493	163105	124890	448463	677562
488893	968277	154571	165413	9859	112092	553475	869328	885439	475340	598622	391932	539477	54218	369430	570546
Bein	bereit	Besitz	Beule	Birne	Boden	Brot	Busch	Dank	denn	Docht	Duft	Durst	ein	einmal	einzel
551904	578884	708129	343721	310313	792094	497819	762015	328439	596164	578754	551017	303040	825829	431497	686943
495973	501884	737014	92280	251907	386655	903955	272409	902289	761556	245698	79474	666701	190206	525643	555035
Ente	Erde	erneut	Erz	falsch	Fell	Filter	Flug	frei	Fuhre	Garn	gefeit	Geist	genehm	Gesetz	gewagt
313203	545038	511851	416671	196973	348971	541849	456940	206299	511274	472166	541510	527355	501389	454812	533659
417230	770853	122091	220073	799190	316190	323682	791873	801774	645869	727378	242709	285853	468831	433707	822890
gleich	Graf	Gruppe	halb	Happen	Hefe	herb	Heu	hint	hold	Idee	intern	jung	Karte	Kind	Klette
503130	493004	469414	502542	373529	286658	567975	411534	769872	587632	365762	500126	454879	504286	981432	656819

Aufgabe 2:

1. Nehme 120 Karten.
2. Bilde 6 Stapel à 20 Karten.
3. Sortiere 1. Stapel nach blauen Zahl. Notiere Zeit!
4. Nehme 2. Stapel dazu und sortiere nach blauen Zahlen.
5. etc.
6. Stelle die jeweils benötigte Zeit graphisch dar.



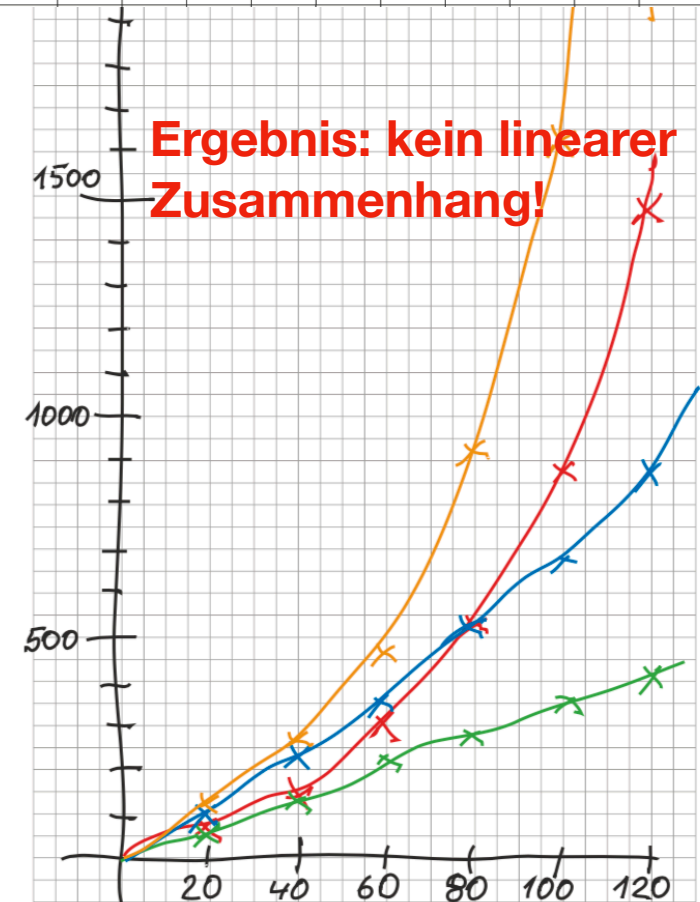
Aufgabe 1: Warm Up!

1. Nehme ca. 60 Karten
2. Sortiere nach blauen Zahlen

187650	302782	503710	861867	212088	522107	38897	955099	96588	424261	747690	44948	581176	133986	467199	145423
aber	Ablage	absurd	Acker	Anbau	Anlage	Anteil	Arbeit	Aufruf	auf tun	Ausland	Auslauf	Auster	Ausweg	Bank	Beere
276637	338210	425565	471452	544520	582278	260249	562184	444082	409121	559505	495493	163105	124890	448463	677562
488893	968277	154571	165413	9859	112092	553475	869328	885439	475340	598622	391932	539477	54218	369430	570546
Bein	bereit	Besitz	Beule	Birne	Boden	Brot	Busch	Dank	denn	Docht	Duft	Durst	ein	einmal	einzel
551904	578884	708129	343721	310313	792094	497819	762015	328439	596164	578754	551017	303040	825829	431497	686943
495973	501884	737014	92280	251907	386655	903955	272409	902289	761556	245698	79474	666701	190206	525643	555035
Ente	Erde	erneut	Erz	falsch	Fell	Filter	Flug	frei	Fuhre	Garn	gefeit	Geist	genehm	Gesetz	gewagt
313203	545038	511851	416671	196973	348971	541849	456940	206299	511274	472166	541510	527355	501389	454812	533659
417230	770853	122091	220073	799190	316190	323682	791873	801774	645869	727378	242709	285853	468831	433707	822890
gleich	Graf	Gruppe	halb	Happen	Hefe	herb	Heu	hinter	hold	Idee	intern	jung	Karte	Kind	Klette
503130	493004	469414	502542	373529	286658	567975	411534	769872	587632	365762	500126	454879	504286	981432	656819

Aufgabe 2:

1. Nehme 120 Karten
2. Bilde 6 Stapel à 20 Karten
3. Sortiere 1. Stapel nach blauen Zahl. Notiere Zeit!
4. Nehme 2. Stapel dazu und sortieren nach roten Zahlen.
5. etc.
6. Stellen die jeweils benötigte Zeit graphisch dar.

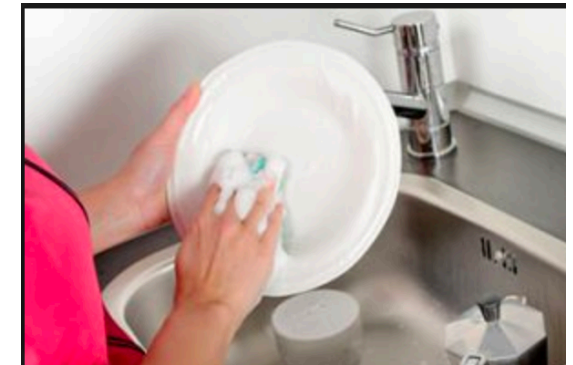


Die Problemgröße

Ein wichtiger Vorgang bei der Lösung von Aufgaben aus der Informationstechnik ist das Bestimmen der sogenannten Problemgröße. Sie stellt ein Maß dar, wie schwierig bzw. umfangreich die Aufgabe bzw. das Problem ist. Daher hängt von der Problemgröße entscheidend der zu leistende Aufwand ab.

Beispiele:

1. Anzahl der Städte auf Landkarte.
2. Anzahl der Teller, die zu spülen sind.
3. Anzahl der zu sortierenden Karten. Aber auch die zu sortierenden Elemente (100-stellige Zahlen sind schwerer zu sortieren als 6-stellige Zahlen!). Was ist hier jetzt die Problemgröße?



Von der Problemgröße hängt der Aufwand zum Erfüllen der Aufgabe ab!



Normalerweise geht man von einem **linearen Verlauf** aus: Zum Spülen von 100 Tellern benötigt man 100-mal so viel Zeit wie zum Spülen eines Tellers.

Beim Sortieren haben wir **keinen linearen Verlauf**: Für die Sortierung von **40 Karten** braucht man i.d.R. knapp **dreimal so lange** wie für **20 Karten**, nicht die doppelte Zeit.

Aufgabe: Was ist der entscheidende Unterschied zwischen dem Sortieren von Karten und dem Spülen von Tellern?





Normalerweise geht man von einem **linearen Verlauf** aus: Zum Spülen von 100 Tellern benötigt man 100-mal so viel Zeit wie zum Spülen eines Tellers.

Beim sortieren haben wir einen **exponentiellen Verlauf**: Für die Sortierung von **40 Karten** braucht man i.d.R. knapp **dreimal so lange** wie für **20 Karten**, nicht die doppelte Zeit.



Aufgabe: Was ist der entscheidende Unterschied zwischen dem Sortieren von Karten und dem Spülen von Tellern?

Lösung:

Dahinter steckt das Grundprinzip der Abhängigkeit und Unabhängigkeit und von „divide et impera“ oder zu Deutsch „teile und herrsche“!

Grundlegende Methode der Informatik: „**divide et impera**“ oder zu **Deutsch** „**teile und herrsche**“.

Das Prinzip „divide et impera“

Sehr häufig hat man im Leben Probleme zu lösen, die zu unüberschaubar und groß sind, um sie in einem Ansatz zu lösen. Vielmehr teilen wir das Gesamtproblem auf in mehrere, handhabbare Stücke, die wir lösen. Die Teillösungen werden danach nur noch zusammengefasst.

Dieses Prinzip wird auch in der Informatik sehr stark verwendet: Ein Programm zerlegt die gestellte Aufgabe zunächst in mehrere kleinere Einheiten, genannt Teilprobleme (divide = teile), und weist danach andere Programme an, diese zu lösen (impera = herrsche, befehlige). Dabei ist sehr wichtig, dass die Teilprobleme unabhängig voneinander gelöst werden können, denn sonst müssten die Programme miteinander kommunizieren, unter Umständen auf Lösungen voneinander warten, was den Aufwand wiederum sehr erhöht.



Grundlegende Methode der Informatik: „**divide et impera**“ oder zu **Deutsch** „**teile und herrsche**“.

Die Aufteilung hängt vom Problem ab:

1. Problem „100 Teller waschen“ → direktes Teilproblem „1 Teller waschen“. Dieses wird 100-mal ausgeführt. Jeder Teller wird **unabhängig** vom anderen gewaschen. Man kann das Problem durch **Parallelisierung** optimieren: Stelle 100 Tellerwäscher an!

Grundlegende Methode der Informatik: „**divide et impera**“ oder zu Deutsch „**teile und herrsche**“.

Die Aufteilung hängt vom Problem ab:

1. Problem „100 Teller waschen“ → direktes Teilproblem „1 Teller waschen“. Dieses wird 100-mal ausgeführt. Jeder Teller wird **unabhängig** vom anderen gewaschen. Man kann das Problem durch **Parallelisierung** optimieren: Stelle 100 Tellerwäscher an!
2. Problem „100 Karten sortieren“ → direktes Teilproblem „1 Karte sortieren“. 1. Karte geht schnell, 2. Karte: größer oder kleiner als 1. Karte?, 3. Karte: $>$ 1. Karte, $>$ 2. Karte? etc. **Der Aufwand zum Lösen der Teilprobleme ist nicht unabhängig von der Problemgröße! Je mehr Karten wir sortieren, desto länger dauert im Mittel das Sortieren einer Karte!**

Grundlegende Methode der Informatik: „**divide et impera**“ oder zu Deutsch „**teile und herrsche**“.

Die Aufteilung hängt vom Problem ab:

1. Problem „100 Teller waschen“ → direktes Teilproblem „1 Teller waschen“. Dieses wird 100-mal ausgeführt. Jeder Teller wird **unabhängig** vom anderen gewaschen. Man kann das Problem durch **Parallelisierung** optimieren: Stelle 100 Tellerwäscher an!
2. Problem „100 Karten sortieren“ → direktes Teilproblem „1 Karte sortieren“. 1. Karte geht schnell, 2. Karte: größer oder kleiner als 1. Karte?, 3. Karte: $><1$. Karte, $><2$. Karte? etc. **Der Aufwand zum Lösen der Teilprobleme ist nicht unabhängig von der Problemgröße! Je mehr Karten wir sortieren, desto länger Dauer im Mittel das Sortieren einer Karte!**

Eine der **wesentlichen Aufgaben eines Informatikers** ist, diese Erhöhung des Zeitaufwands zu erforschen und dieses Wissen zu nutzen, um die verwendeten **Programme zu verbessern, so dass auch hohe Problemgrößen noch bewältigt** werden können.

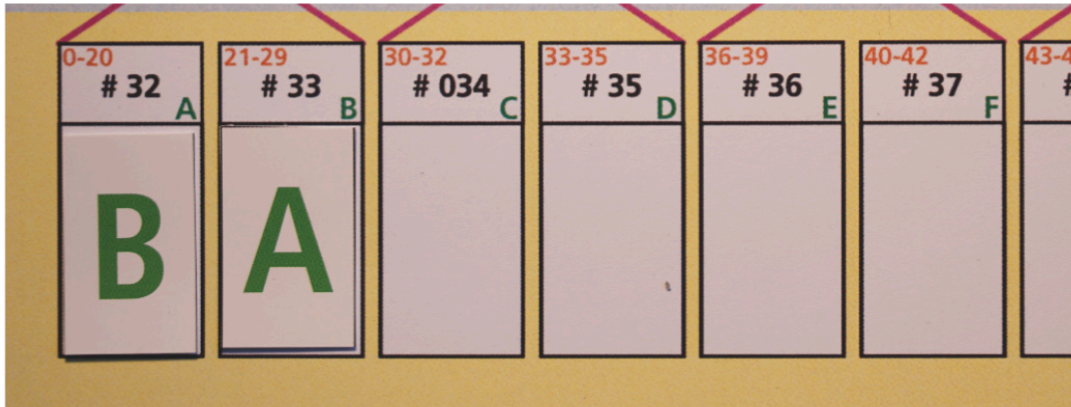
Sortieralgorithmen:

Der **größte Unterschied zwischen Mensch und Maschine** besteht beim Sortieren in der **Speicherung**:

- Der menschliche Kartensortierer besitzt quasi überall „Speicher“ – auf der Hand, als Stapel auf dem Tisch, ausgelegt auf dem Tisch etc.
- Beim Computer ist der Speicher streng in Positionen organisiert und jede Position, genannt **Speicherstelle** oder **Speicheradresse**, kann **genau einen Wert bzw. in unserem Spiel eine Karte** beinhalten.

Computerspeicher für die
Karten aus der
Kopiervorlage →

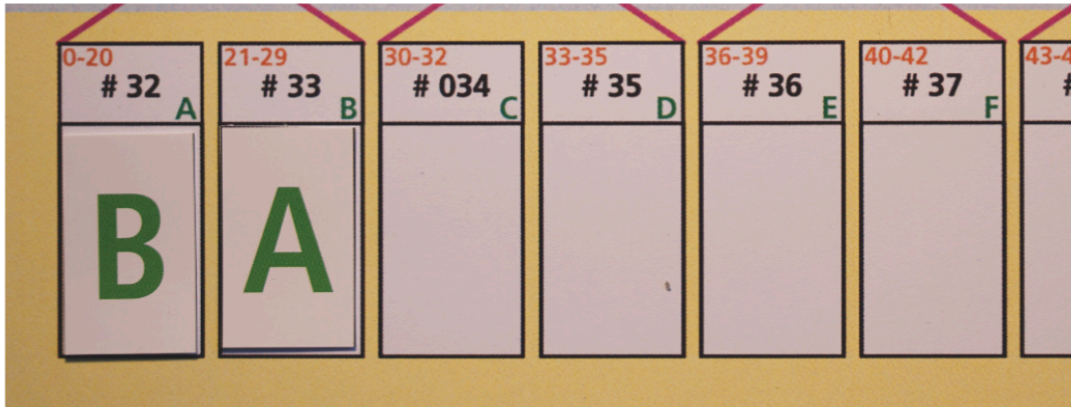




Jede Adresse #XX kann genau **einen Wert** speichern!

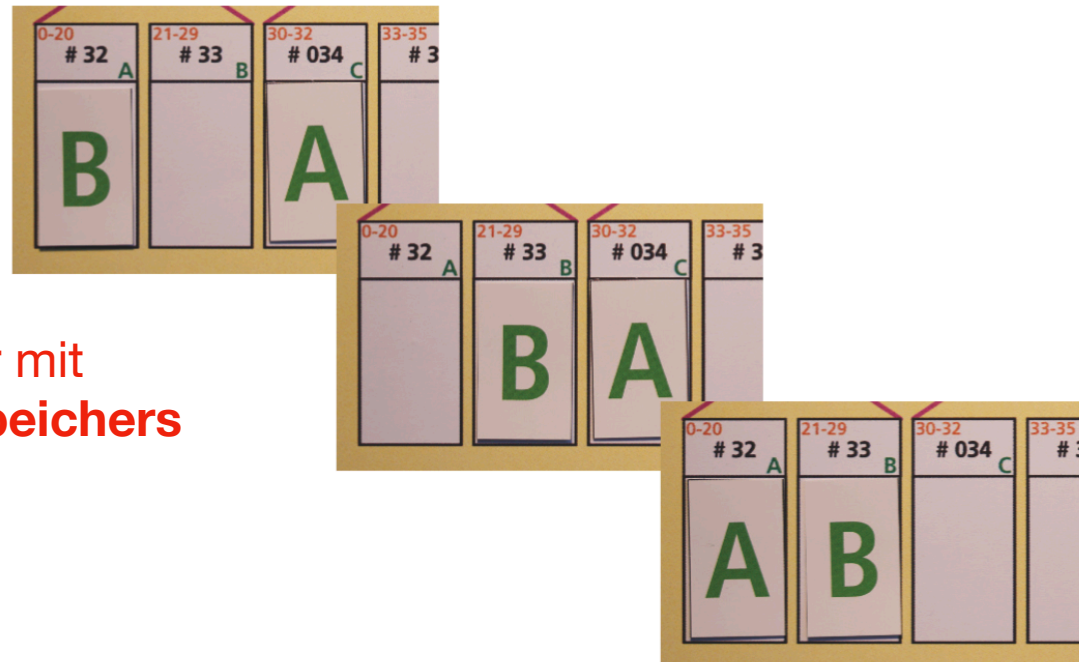
—> Spielregel: Sobald eine Karte auf eine andere Karte gelegt wird, fliegt die unter Karte aus dem Spiel.

1. Lege zwei beliebige Karten auf die Adressen #32 und #33.
2. Tauschen Sie beide Karten aus. **Wichtig: Halten Sie sich an die Spielregeln!**



Jede Adresse #XX kann genau **einen Wert** speichern!
—> **Spielregel 1: Sobald eine Karte auf eine andere Karte gelegt wird, fliegt die unter Karte aus dem Spiel.**

1. Lege zwei beliebige Karten auf die Adressen #32 und #33.
2. Tauschen Sie beide Karten aus. **Wichtig: Halten Sie sich an die Spielregeln!**



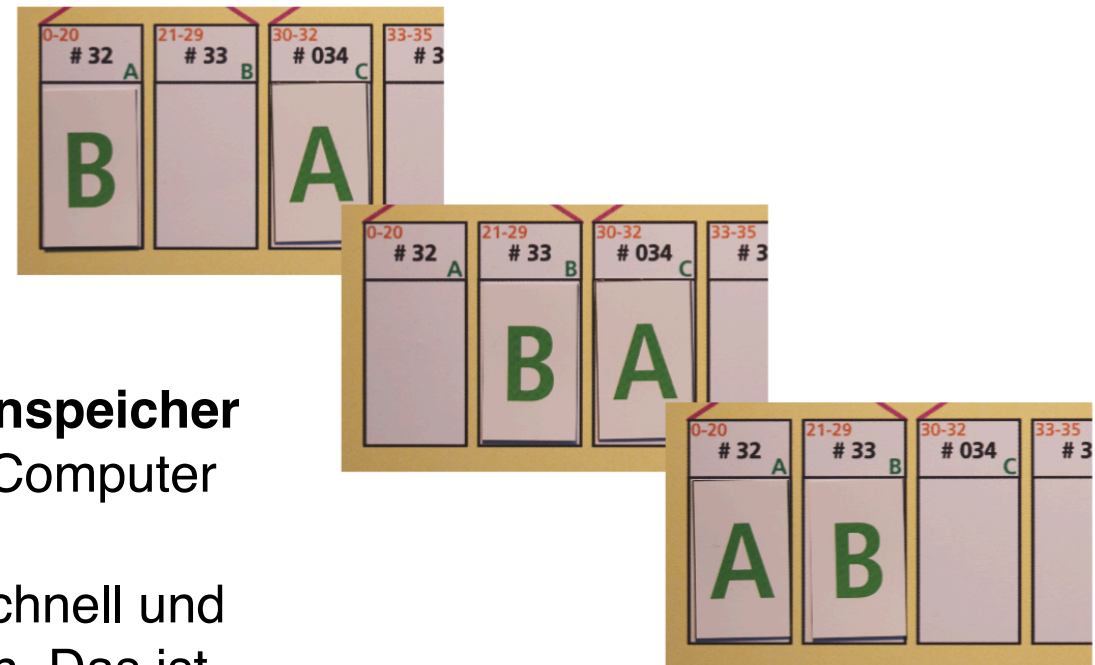
Ein Kartenaustausch ist nur mit Hilfe eines **zusätzlichen Speichers** möglich! Der sogenannte **Zwischenspeicher**.

Ein Kartenaustausch ist nur mit Hilfe eines **zusätzlichen Speichers** möglich! Der sogenannte **Zwischenspeicher**.

Für viele Vorgänge wird ein **Zwischenspeicher** benötigt. Dafür besitzen die meisten Computer auch spezielle Vorkehrungen – **Speicheradressen**, die besonders schnell und komfortabel abgefragt werden können. Das ist der sogenannte Arbeitsspeicher.

—> **Spielregel 2: Maximal eine Karte in der Hand halten, während Sie die anderen Karten im Speicher verschieben.**

1. Lege 6 beliebige Karten auf die Adressen #32 und #37.
2. Tauschen Sie beide Karten aus. **Wichtig: Halten Sie sich an die Spielregeln!**

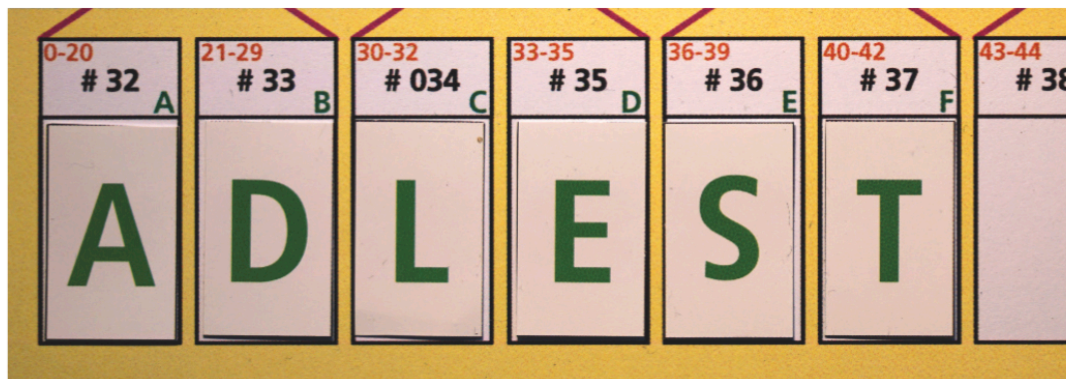
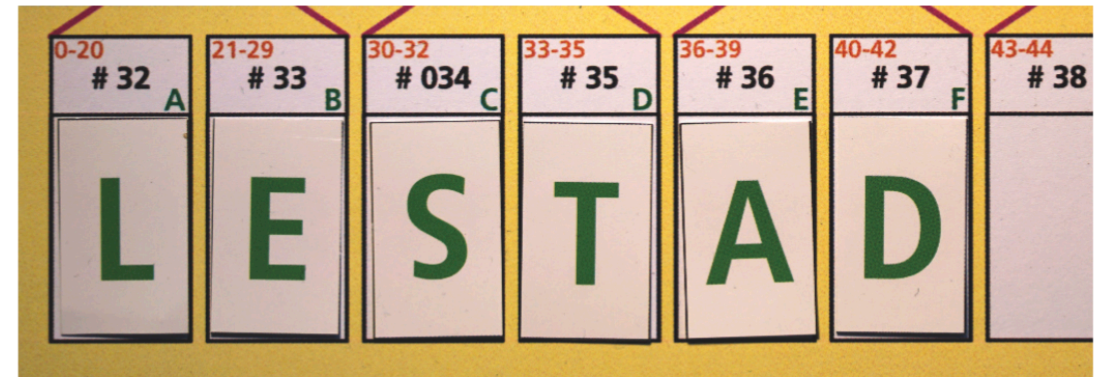




Aufgabe:

1. Lege 6 beliebige Karten auf die Adressen #32 und #37.
2. Folge soll alphabetisch sortiert werden.
 1. Nehme A in den Zwischenspeicher und schiebe L,E,S und T eine Position nach rechts. Setze A auf #32.
 2. Nehme D in den Zwischenspeicher ...

zu Beginn



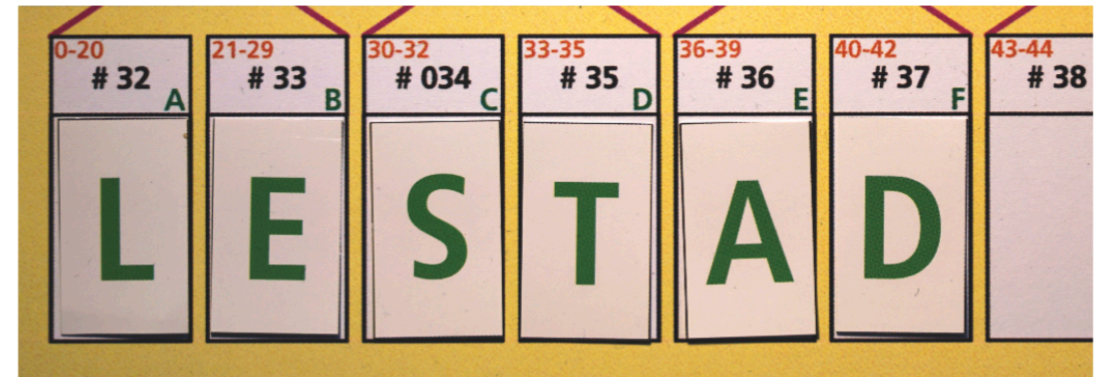
nach dem 2. Schritt



Aufgabe:

1. Lege 6 beliebige Karten auf die Adressen #32 und #37.
2. Folge soll alphabetisch sortiert werden.
 1. Nehme A in den Zwischenspeicher und schiebe L,E,S und T eine Position nach rechts. Setze A auf #32.
 2. Nehme D in den Zwischenspeicher ...

zu Beginn



nach dem letzten Schritt

Zwei Erkenntnisse haben wir hier genutzt, die wir als Menschen sofort haben, die der Computer aber nicht ohne Weiteres besitzen kann:

1. A ist im Speicher die Karte mit dem alphabetisch kleinsten Buchstaben.
2. Die Karte mit dem kleinsten Buchstaben liegt auf Position #36.

Woher wissen wir das? Weil wir natürlich alle Karten gleichzeitig sehen und bereits unser Unterbewusstsein spontan die Situation für uns analysiert.

Zwei Erkenntnisse haben wir hier genutzt, die wir als Menschen sofort haben, die der Computer aber nicht ohne Weiteres besitzen kann:

1. A ist im Speicher die Karte mit dem alphabetisch kleinsten Buchstaben.
2. Die Karte mit dem kleinsten Buchstaben liegt auf Position #36.

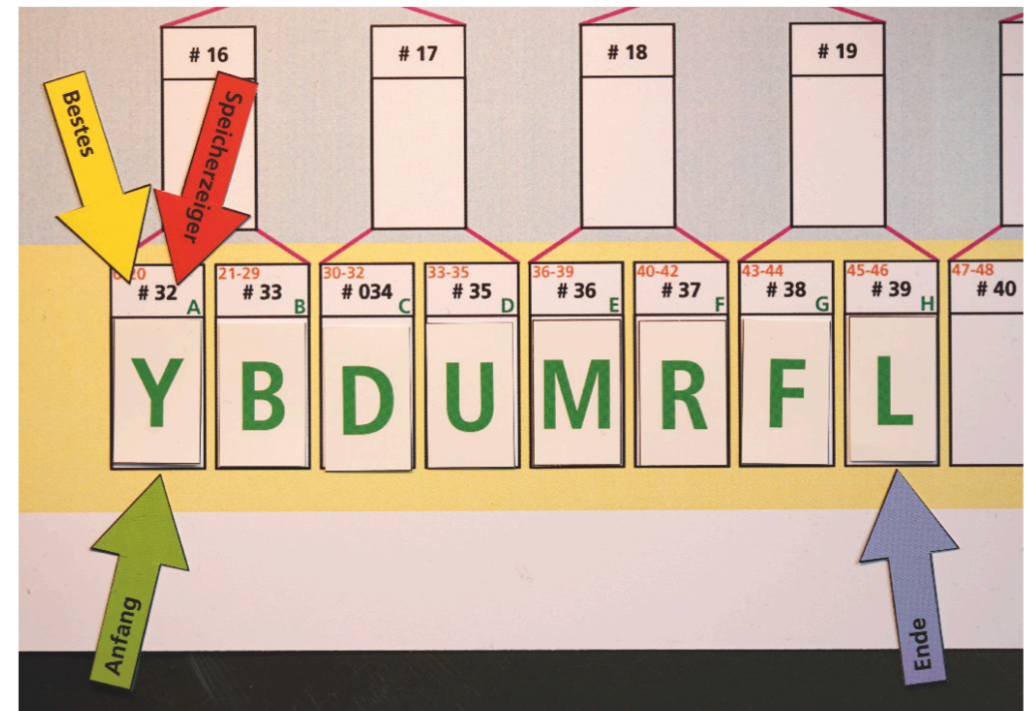
Woher wissen wir das? Weil wir natürlich alle Karten gleichzeitig sehen und bereits unser Unterbewusstsein spontan die Situation für uns analysiert.

Ein **Computer** kann zwar auf seinen gesamten Speicher zugreifen, aber im Normalfall **immer nur gleichzeitig eine Adresse anschauen**.

—> **Spielregel 3: Wenn wir eine Speicherstelle auslesen wollen, muss immer ein Pfeil auf die entsprechende Adresse zeigen. Alle anderen Adressen gelten als unsichtbar.**



- > Spielregel 1: Sobald eine Karte auf eine andere Karte gelegt wird, fliegt die unter Karte aus dem Spiel.
- > Spielregel 2: Maximal eine Karte in der Hand halten, während Sie die anderen Karten im Speicher verschieben.
- > Spielregel 3: Wenn wir eine Speicherstelle auslesen wollen, muss immer ein Pfeil auf die entsprechende Adresse zeigen. Alle anderen Adressen gelten als unsichtbar.

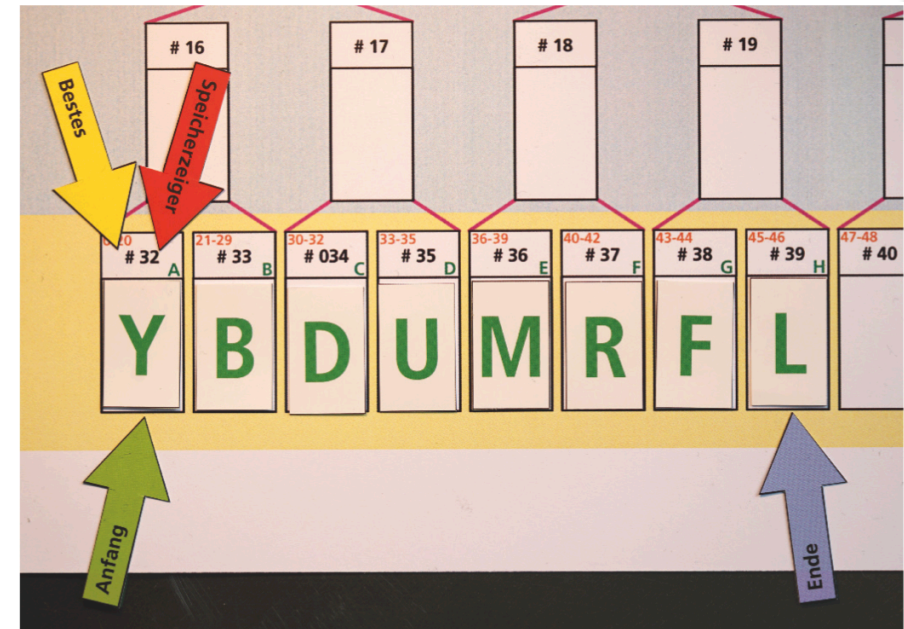




Sortieralgorithmus „**Selection-Sort**“ —>
Unterteilung in kleinere Probleme:

Sortieren von **n** Karten:

- Sortiere die Karte mit dem kleinsten Buchstaben an den Anfang
- Sortiere die **restlichen n – 1** Karten



Aufgabe:

Konkretisieren Sie die Vorschrift! Wie geht es nach Schritt 2 weiter?

Anfang und Ende zeigen auf die erste bzw. die letzte zu sortierende Karte

0. Setze Speicherzeiger und Bestes auf die erste zu sortierende Karte

1. Verschiebe Speicherzeiger um eine Position nach rechts

2. Kommt der Buchstabe unter Speicherzeiger alphabetisch früher als der Buchstabe unter Bestes ?

Falls JA: Verschiebe Bestes auf die Position von Speicherzeiger



Sortieralgorithmus „**Selection-Sort**“ —> **Unterteilung in kleinere Probleme:**

Sortieren von **n** Karten:

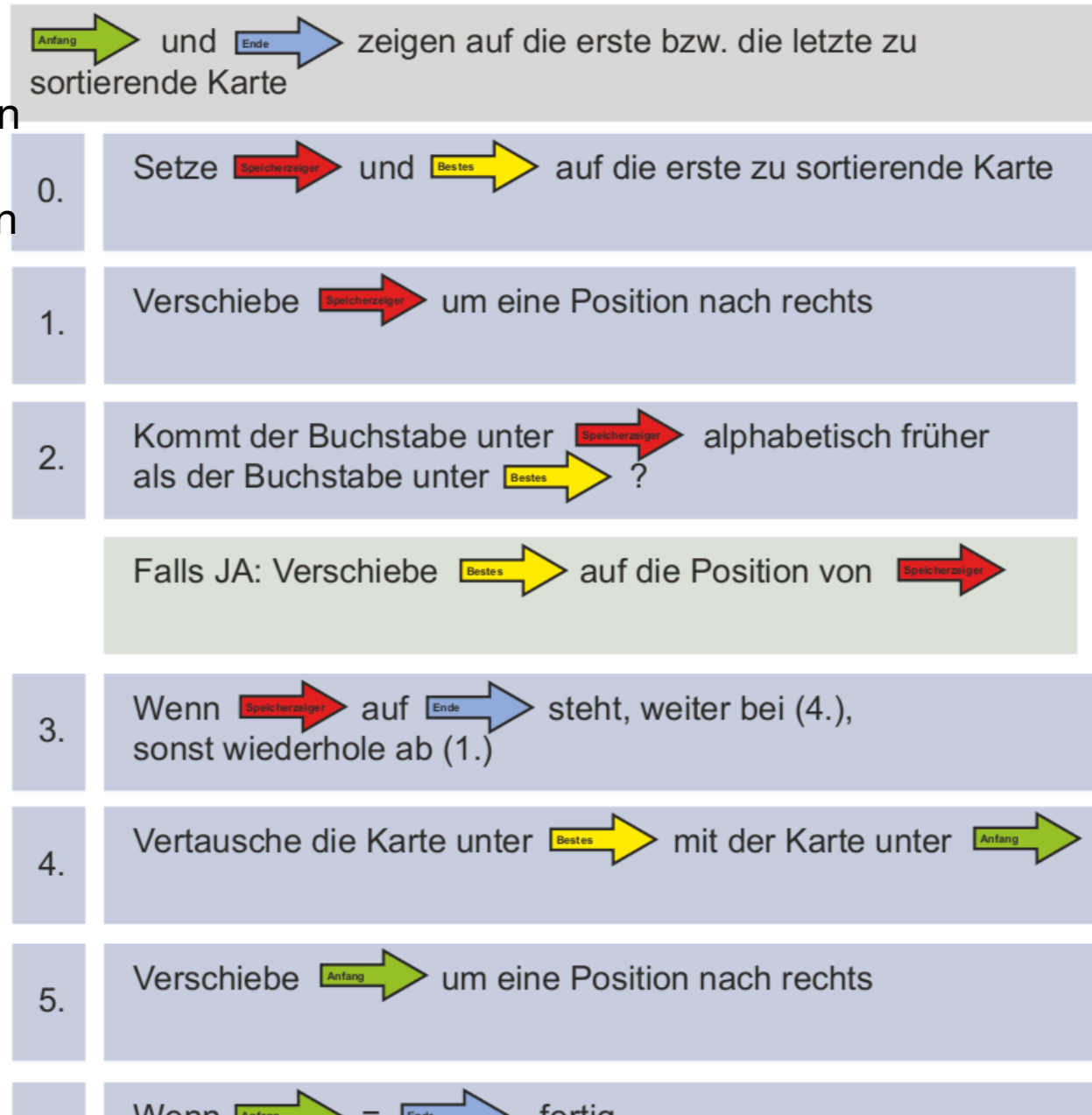
1. Sortiere die Karte mit dem kleinsten Buchstaben an den Anfang
2. Sortiere die **restlichen n – 1** Karten

Lösung:

In den Anweisungen (1.) bis (3.) wird der kleinste Buchstabe gesucht und gefunden. Dieser muss ganz an den Anfang und wird daher getauscht mit der Karte, die den Platz dort belegt (4.).

Nun weiß man, dass die Karte an der ersten Position korrekt liegt, dass diese also nie mehr verschoben werden muss: Sie gehört nicht mehr zum zu sortierenden Bereich. Daher rücken wir bei (5.) den Anfangszeiger um eine Position nach rechts und wiederholen das Sortieren so lange, bis die unsortierte Folge nur noch aus einer einzelnen Karte besteht.

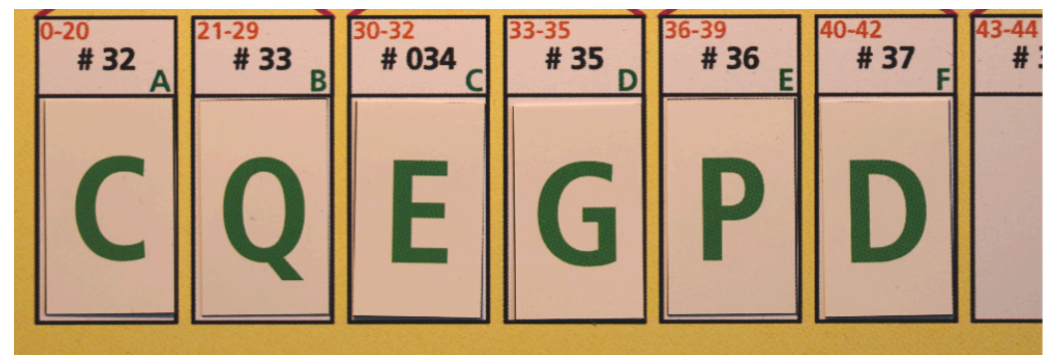
Eine Karte ist aber sowieso immer sortiert, daher sind wir fertig!



Aufgaben:

1. „Begreifen“ Sie den Algorithmus, indem Sie mehrere zufällige Folgen von Karten sortieren. Nehmen Sie hierfür auch gerne die Karten mit den echten Zahlen.

Sie können sowohl die rote als auch die blaue Reihe nutzen – wenn Sie das abwechselnd tun, sparen Sie sich das Mischen zwischendurch.



2. Es gibt noch viele andere Sortieralgorithmen (Bubble-Sort, Quick-Sort, Mergesort etc.)! Bilden Sie Gruppen à 3-4 Personen:

Begreifen Sie den jeweiligen Algorithmus und bereiten Sie ihn **bis Donnerstag den 27.09** so vor, dass sie den Algorithmus Ihren Klassenkameraden erklären können! Bereiten Sie dafür ein Handout, eine Präsentation und/oder einen Tafelvortrag vor!



2. Es gibt noch viele andere Sortieralgorithmen (Bubble-Sort, Quick-Sort, Mergesort etc.)! Bilden Sie Gruppen à 3-4 Personen:

Begreifen Sie den jeweiligen Algorithmus und bereiten Sie ihn **bis Donnerstag** den **27.09** so vor, dass sie den Algorithmus Ihren Klassenkameraden erklären können! Bereiten Sie dafür ein Handout, eine Präsentation, YouTube Clip, Plakat und/oder einen Tafelvortrag vor! Für die Präsentation haben Sie maximal 30 Minuten.

Gruppen à 3-4 Personen:

- 1. Bubble Sort: Mohammed, Omar, Michael, Ha, Chi**
- 2. Quicksort: Natalia, Adinda, Alex**
- 3. Mergesort: Bahaa, Ramadhan, Fadhyl, Samuel**
- 4. Shellsort: Aryo, Amr, Bimo**
- 5. Heapsort: Erniesta, Alisa, Diana, Dinda, Rahima**

Falls Sie den Beamer nutzen wollen, bringen Sie die Präsentation auf einem USB Stick mit.

Alternativ können Sie die Präsentation über einen stationären Computer oder über Ihren eigenen Laptop (VGA Anschluss) ausführen.



Worauf Sie bei einer guten Präsentation achten sollten:

- 1. Ist die Fragestellung gut erklärt?**
- 2. Ist die Gliederung des Vortrags nachvollziehbar?**
- 3. Kommen Sie mit der Zeit aus? Proben Sie den Vortrag mindestens 2 Mal!**
- 4. Kommen alle Beteiligten zu Wort? Ist ersichtlich, wer woran gearbeitet hat?**
- 5. Sind alle Quellen angegeben?**
- 6. Das Thema muss 100%ig verstanden sein von allen Gruppenmitgliedern.**
- 7. Bei Gruppenarbeiten müssen die Aufgaben (Recherche, PowerPoint/Tafelbild, Moderation etc.) klar verteilt werden.**
- 8. Das Ziel des Vortrags muss klar erkennbar sein.**



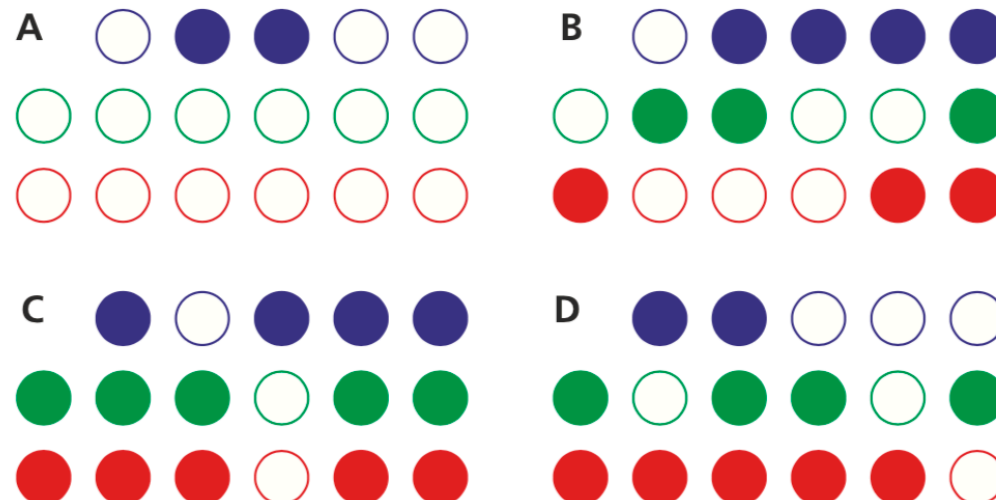


Computer rechnen nicht mit 10 Fingern, sondern mit zwei!

Warum? Eine automatische Schalteinrichtung (sogn. Relais) hat **zwei Schaltzustände**: an/aus. Daher waren die ersten Computer – der aus diesen Bauteilen bestanden – automatisch auf ein Zahlensystem festgelegt.



Ist das eine Uhr?

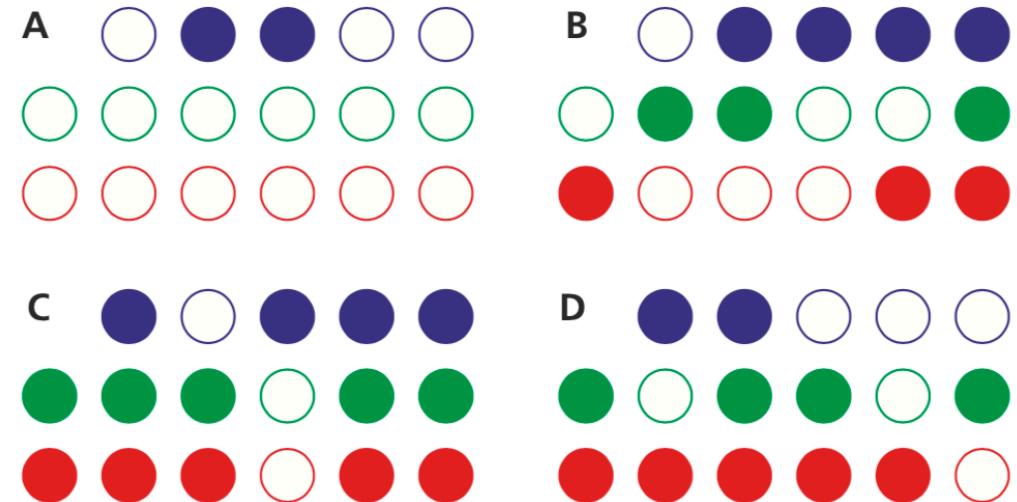


<http://www.abenteuer-informatik.de/bu.html>



<http://www.abenteuerinformatik.de/bu.html>

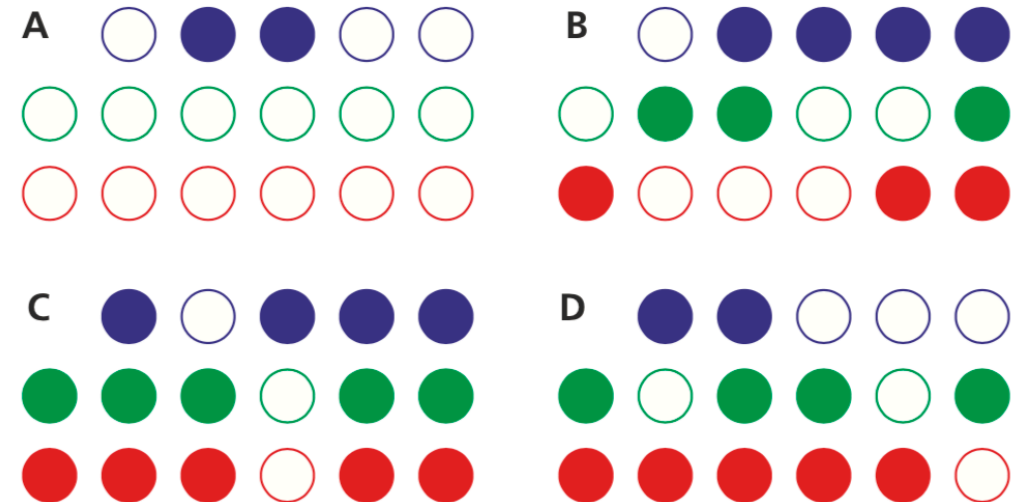
- roten Punkte wechseln ihren Zustand im Sekundentakt,
- die grünen minütlich
- die blauen jede Stunde.



Ist das eine Uhr?

<http://www.abenteuer-informatik.de/bu.html>

- roten Punkte wechseln ihren Zustand im Sekundentakt,
- die grünen minütlich
- die blauen jede Stunde.



Ist das eine Uhr?

Eine Uhr **zählt** allerdings Sekunden, Minuten und Stunden, und dieser **Zählvorgang** wird nicht auf den ersten Blick klar.



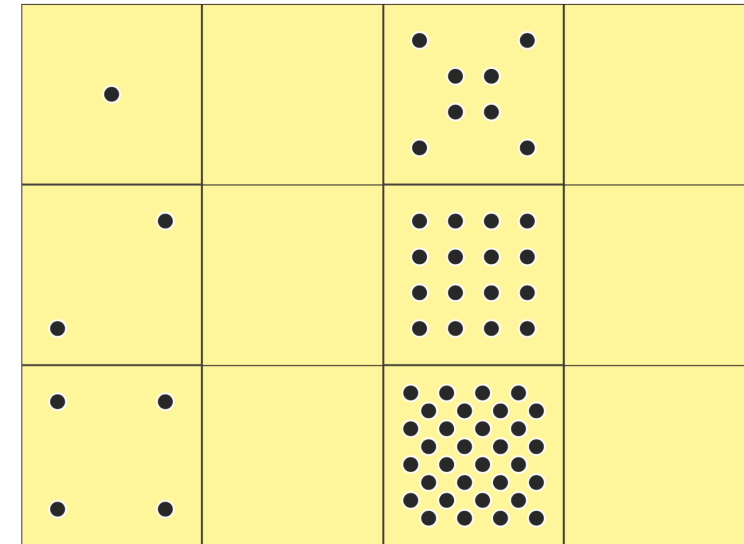
Aufgabe:

1. Versuchen Sie, die Karten nacheinander so auf den Tisch zu legen, dass diese insgesamt 13 Punkte zeigen.
2. Versuchen Sie auch andere Zahlen darzustellen.
3. Bekommen Sie 25 Punkte, 44 und auch 68?



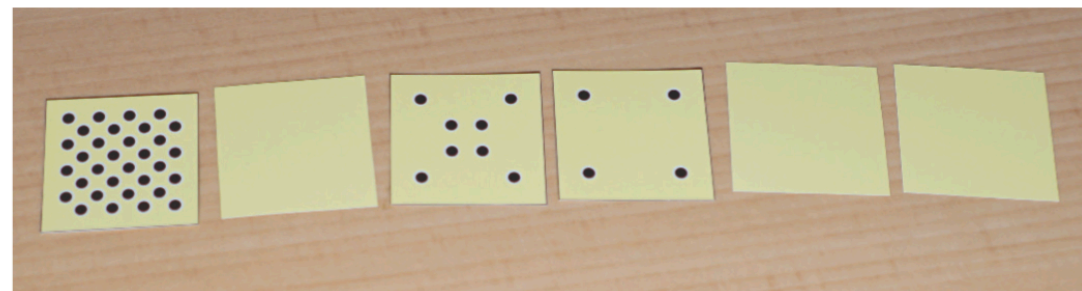
Aufgabe:

1. Versuchen Sie, die Karten nacheinander so auf den Tisch zu legen, dass diese insgesamt 13 Punkte zeigen.
2. Versuchen Sie auch andere Zahlen darzustellen.
3. Bekommen Sie 25 Punkte, 44 und auch 68?



Probleme bei 64 und mehr, da nur 63 Punkte vorhanden sind!

44 Punkte



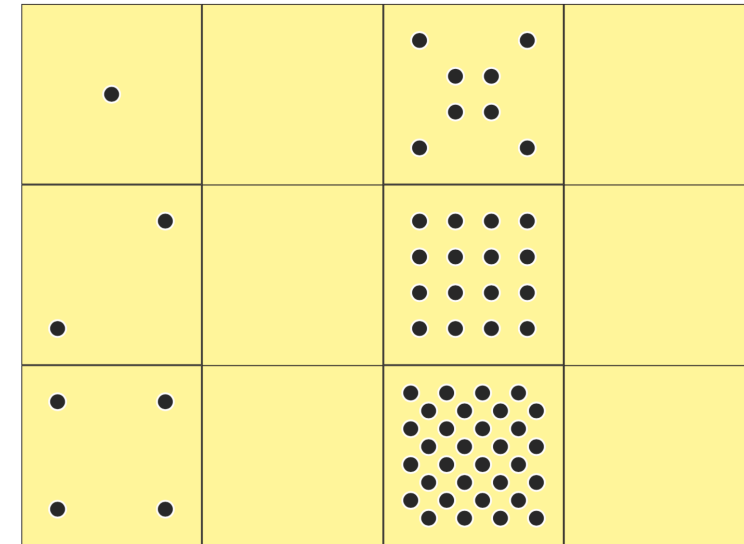


Frage:

1. Können wir alle Zahlen zwischen 1 und 63 darstellen?

Aufgabe:

1. Sortiere die gelben Karte nach der Zahl der aufgedruckten Punkte.
2. Lege nacheinander alle Zahlen 1,2,3 etc.
3. Achten darauf, wie oft Sie die einzelnen Karten umdrehen!



44 Punkte

