1#Involution

1.1 Didaktischer Kommentar

Dominic Harion & Ann Kiefer

„Mit Big Data und den dazugehörigen Algorithmen erleben wir eine verstärkte Rückkehr der Mathematik und naturwissenschaftlicher Methoden in die Organisation des Sozialen", so der Befund Felix Stalders zu automatisierten Entscheidungssystemen in unseren digitalen Kulturen (Stalder, 2017): Algorithmen sind, abseits der Bedeutung, die ihnen in der Informatik zukommt, ganz allgemein zunächst einmal „Handlungsvorschriften zur Lösung eines Problems oder einer Klasse von Problemen und bestehen aus endlich vielen, wohldefinierten Einheiten" – auch Gebrauchsanweisungen für Maschinen und Kochrezepte sind letztlich Algorithmen. Kenntnisse darüber, was solche Algorithmizität in unserer Umwelt bedeutet und worin sie wirkt, *insbesondere* in ihrer Verschränkung von Technologien mit unseren Alltagswelten, gehören nun ebenso zu einem fundierten Bildungsanspruch wie Fertigkeiten darin, sie zu modellieren, anzuwenden und gegebenenfalls zu modifizieren. Algorithmen sind zunächst nichts spezifisch Technologisches, gar Technisches oder Digitales. Sie können (und sollten) in solchen spezifischen Ausformungen aber auch nicht mehr aus unseren Infrastrukturen getilgt werden und übernehmen dabei als entlastende Institutionen nicht zuletzt komplexe Prozesse.

Ein solcher Einsatzbereich, in dem Algorithmen Bestandteil von sozialen Organisationsprozessen sind, findet sich in Systemen zur Kartographierung von und Mobilitätsorganisation in Topographien, also z.B. Plänen des öffentliche Personennahverkehrs und Mobilitäts-Apps. Der „Algorithmus des kürzesten Weges" in einem Grafen, in Fachkreisen als *Breadth-first search-Algorithmus* oder *Dijkstra-Algorithmus* (in solchen Fällen, in denen die Kanten des Grafen Gewichte haben) bekannt, stellt damit ein sehr alltagsnahes und zugleich gut selbständig erforschbares Fallbeispiel zur Umsetzung im Unterricht dar. Im Rahmen des hier vorgestellten Moduls wird ein spielerischer und kollaborativer Zugang zu diesem Algorithmus über *Involution*©, ein an der Universität Luxemburg von Hugo Parlier und Bruno Teheux entwickeltes mathematisches Rätsel-Spiel, gewählt. Das Erlernen algorithmischer Modellierungen über spielerische und kompetitive Verfahren wird bereits seit längerem erfolgreich anhand von „Zauberwürfeln" (*Rubik's Cube*) erprobt und diskutiert (vgl. etwa Lakkaraju et al., 2022), die Lernzugänge an der Schnittstelle von Mathematik und Informatik eröffnen (vgl. etwa Joyner 2008; Agostinelli et al. 2019). Während anhand solcher Würfel dreidimensionale Muster veranschaulicht werden können, gestattet es *Involution*©, algorithmische Modelle im zweidimensionalen Raum nachzuvollziehen und zu klären. Es ist damit geeignet, über mathematische Modellbildung an lebensweltlichen Problemzusammenhängen einen transdisziplinären Lernhorizont in Digital Sciences zu eröffnen: Das Spiel mit *Involution*© wird mit dem Auftrag parallelisiert, den kürzesten Weg in einem U-Bahn-System herauszufinden, den die Schüler*innen bereits aus einem Digital Sciences-Kurs kennen (Digital Sciences, 2021). Solche Aufgaben zu *Shortest-Path-Problemen* wurden bereits mit Kindern und Jugendlichen verschiedener Altersstufen explorativ bearbeitet und zur Konkretisierung abstrakter formaler Modelle verwendet (vgl. Gibson, 2012) und lassen sich entsprechend in gestaffelten Schwierigkeitsgraden für differenzierendes Unterrichten einsetzen.

Didaktisch ist das Modul an den beiden Prinzipien des *problemlösenden* und *kooperativen Lernens* orientiert. Die Lernenden werden mit einer Problemstellung konfrontiert, zu deren Lösung eine Spielvorlage bereitgestellt und grundlegende Arbeitsanweisungen ausgegeben werden.

Damit wird ein zielgerichteter kognitiver Prozess angeregt, um eine gegebene Situation in eine Zielsituation umzuwandeln, ohne eine offensichtliche Methode zur Lösung vorzugeben, wobei kreatives und kritisches Denken angesprochen werden (vgl. Mayer & Wittrock, 2006). So wird den Schüler*innen nicht zuletzt ein Perspektivenwechsel auf die MINT-Fächer ermöglicht: Für viele Jugendliche geht es in diesen Fächern darum, die „einzig richtige" Lösung zu finden und dies möglichst schnell, welcher Zugang jedoch nicht der Realität der Bezugswissenschaften entspricht. Wissenschaftler*innen aus den MINT-Fächern haben meistens keine klare Idee, in welche Richtung sie nach einer Lösung suchen müssen – dieses Mindset kommt mitunter in den schulischen MINT-Fächern zu kurz.

Rätsel-Spiele wie Involution© bieten so auch den perfekten Rahmen, um die Resilienz der Schüler zu trainieren. „Resilienz bezieht sich auf die affektive Fähigkeit der Schüler*innen, mit Hindernissen und negativen Situationen im Lernprozess umzugehen und diese zu überwinden, indem sie diese negativen Situationen in Situationen umwandeln, die sie unterstützen" (Hutauruk & Priatna, 2017). Schüler*innen, die nicht gewohnt sind, mit frustrierenden Lernsituationen und Misserfolgen umzugehen, betrachten diese als sehr negativ. Werden Schüler:innen aber daran gewöhnt, so hat diese Resilienzerfahrung durchaus positive Folgen für das Studium und die spätere Arbeitswelt der Jugendlichen (Hutauruk & Priatna, 2017).

Die Lösung der Aufgabe erfolgt nicht in Einzelarbeit, sondern in Dyaden oder Kleingruppen und wird kompetitiv (aber nicht konkurrierend) aufgebaut. Neben dem spielerisch-motivationalen Anreiz werden damit die Modellierung und Verbalisierung eigener Lösungswege angeregt und Möglichkeiten des Lernens über Lehren gegeben, indem die Schülerinnen und Schüler ihre Denkprozesse kommunizieren und argumentativ untermauern müssen. Dabei ist sowohl ein Setting denkbar, in dem Lerngruppen homogen nach dem Schwierigkeitsgrad der Aufgabenstellungen zusammengestellt werden wie auch eine Zusammenstellung von heterogenen Lerngruppen, in dem Schüler:innen durch ihre Mitschüler:innen bei dem Erarbeiten der Lösungswege gecoacht werden.

Referenzen:

Agostinelli, Forest, Mavalankar, Mihir, Khandelwal, Vedant, Tang, Hengtao, Wu, Dezhi, Berry, Barnett, Srivastava, Biplav, Sheth, Amit & Irvin. Matthew. *(2021)*. Designing Children's New Learning Partner: Collaborative Artificial Intelligence for Learning to Solve the Rubik's Cube. In *Interaction Design and Children (IDC ,21)*. Association for Computing Machinery, New York, NY, USA, 610–614. https://doi.org/10.1145/3459990.3465175
Digital Sciences. (2021). *Mon monde numérique et moi.* Schüler Achse 1.
Gibson, Paul J. (2012). Teaching graph algorithms to children of all ages. In *Proceedings of the 17th ACM annual conference on Innovation and technology in computer science education (ITiCSE ,12)*. Association for Computing Machinery, New York, NY, USA, 34–39. https://doi.org/10.1145/2325296.2325308
Hutauruk, Agusmanto J.B., & Priatna, Nanang. (2017). Mathematical Resilience of Mathematics Education Students. *J. Phys.: Conf.* Ser. 895 012067. https://iopscience.iop.org/article/10.1088/1742-6596/895/1/012067/pdf
Joyner, David (2008). *Adventures in group theory: Rubik's Cube, Merlin's machine, and other mathematical toys* (2nd ed.). Baltimore : John Hopkins University Press.
Lakkaraju, Kausik, Hassan, Thahimum, Khandelwal, Vedant, Singh, Prathamjeet, Bradley, Cassidy, Shah, Ronak, Agostinelli, Forest, Srivastava, Biplav, & Wu, Dezhi.(2022). ALLURE: A Multi-Modal Guided Environment for Helping Children Learn to Solve a Rubik's Cube with Automatic Solving and Interactive Explanations. (Preliminary Preprint). https://www.aaai.org/AAAI22Papers/DEMO-00182-LakkarajuK.pdf
Mayer, Richard E. & Wittrock, Merlin C. (2006). Problem Solving. In P. A. Alexander & P. H. Winne (Eds.). *Handbook of Educational Psychology* (2nd Ed), 287-303. New York : Routledge.
Rogers, Hartley (1987). *Theory of Recursive Funtions and Effective Computability.* Massachusetts : C1957.
Stalder, Felix. (2017). *Algorithmen, die wir brauchen.* https://netzpolitik.org/2017/algorithmen-die-wir-brauchen/

1.2 Unterrichtsplanung

01 | Thema der Einheit im Gesamtgefüge der Achsen

Modul	Thematische Achsen	Schwerpunkte	Fächerübergreifend mit folgenden Disziplinen
#Involution	Achse 1 Meine digitale Welt und ich!	• Spiele und Algorithmen • Algorithmen des kürzesten Weges	• Mathematik • Geografie
#Climate Killer Internet	Achse 2 Das Internet verstehen: World Wide Web und ich.	• Internet und Klima • Urteilskompetenz	• VIESO • Geografie • Deutsch • Französisch
#Data Viz Superpowers	Achse 3 Do you speak Informatik?	• Verschiedene Formen der Data Visualisation • Manipulation von Grafiken	• Kunstunterricht • Mathematik
#Discover Life on Mars with a Rover	Achse 5 Der Roboter, ein Partner im guten und im schlechten Sinne?	• Programmieren in Scratch • Educational Robotics	• VIESO
#Pupils vs Machine	Achse 6 Gibt es eine Maschine, die so intelligent ist wie ich?	• Basisfunktionsweise einer KI	• Mathematik • VIESO

Da die Module unabhängig voneinander funktionieren, kann dieses Modul im Unterricht behandelt werden, ohne dass die anderen Module behandelt werden müssen.

Dieses Modul wurde in Zusammenarbeit mit Hugo Parlier und Bruno Teheux von der Mathematischen Fakultät der Universität Luxemburg ausgearbeitet.

02 | Bedingungsanalyse

1. Zielgruppe: 7e-5e klassischer und allgemeiner Sekundarunterricht
2. Raum: Es sind keine besonderen Räumlichkeiten vorzusehen
3. Benötigtes Material:
   - Das Spiel Involution©: Das Spiel kann im Centre de documentation pédagogique des IFEN [https://ssl.education.lu/ifen/mh_centre-de-documentation] oder im F.use (Future Space for Education) auf dem Campus Belval ausgeliehen werden.
   - Computer oder Tablets können verwendet werden, um die Anweisungen zu lesen, notwendig ist das aber nicht. Die Anweisungen können auch einfach ausgedruckt werden.
4. Dauer: 2 Unterrichtsstunden

03 | Sachanalyse

Die Hauptidee dieser Stunde besteht darin, das im Unterricht behandelte Thema „Algorithmen" zu konkretisieren, und zwar mit *Involution*©, einem Spiel, das von Hugo Parlier und Bruno Teheux, zwei Mathematikwissenschaftlern der Universität Luxemburg, entwickelt wurde. Durch dieses Spiel werden die Schüler*innen auf spielerische Weise mit mathematischen und algorithmischen Konzepten wie dem Algorithmus des kürzesten Pfads sowie optimalen und nicht-optimalen Strategien vertraut gemacht.
Das Spiel besteht aus einer Anordnung von 10 schwarzen und weißen Ringen. Eine kreisförmige Kurbel wird auf das Spielbrett gesetzt und durch eine Kurbelbewegung werden die Positionen der Ringe verändert. Ziel des Spiels ist es, die Position der Ringe so zu verändern, dass man ausgehend von einer Startkonfiguration zu einer bestimmten Zielkonfiguration gelangt.

04 | Didaktische Analyse

a. Angestrebte Lernziele und Kompetenzen

Angestrebte Kompetenzen der Achse 1: Meine digitale Welt und ich!

- WISSEN: Grundkenntnisse des allgemeinen digitalen und computerbezogenen Vokabulars, der Kommunikation zwischen Mensch und Computer (Algorithmus) und des Abstraktionsprinzips bei der Problemlösung.
- FACHKOMPETENZ: Zerlegung eines Problems zum Zweck der Problemlösung, Konzeption und Darstellung eines Algorithmus, Entwicklung und Anwendung eines einfachen Algorithmus mit Karten und einem Programmablauf.
- VERHALTEN: Förderung des Bewusstseins für die Bedeutung der Informatik für die täglichen Aufgaben auf der Grundlage der gewählten Tools, der bereitgestellten Informationen und der erzielten Ergebnisse.

Lernziele aus dem Medienkompass[1]

- MK2 – Kommunikation und Zusammenarbeit: 2.1 Mit anderen zusammenarbeiten
- MK3 – Erstellen von Inhalten: 3.3 Modellieren, strukturieren und codieren

b. Didaktische Relevanz und Begründung

In diesem Modul sollen das Konzept und die Funktionsweise von Algorithmen anhand eines Spiels verständlich gemacht werden. Eine Besonderheit des Moduls ist, dass keine elektronischen Technologien, weder Computer noch Tablets, verwendet werden. Mit solchen „Unplugged"-Aktivitäten sollen Schüler*innen in die Welt der Informatik eingeführt werden, um „die technologische Komplexität beim Lernen zu beseitigen und die Grundbegriffe der Informatik zu vermitteln" (INRIA, 2020). Der Zugang wird dadurch erleichtert und verdeutlicht einen wesentlichen Aspekt der digitalen Arbeit: Digitale Kulturen haben in erster Linie

[1] https://www.edumedia.lu/medienkompass/medienkompass/

nichts mit Computern im Sinne von elektrischen Maschinen zu tun – das Merkmal der Algorithmizität beispielsweise, das im Mittelpunkt dieses Moduls steht, lässt sich durch die Bildung formaler mathematischer und logischer Modelle beschreiben, die auch mechanisch dargestellt werden können. Um den Informatiker Leslie Lamport zu zitieren: „Die Bedeutung des Nachdenkens und des Aufschreibens vor dem Codieren sollte bereits in der Vorschule gelehrt werden, wird es aber nicht.“ (Han, 2022).

So knüpft das Modul *#Involution* an die in der Grundschule vermittelten Grundlagen des Codierens an und entwickelt sie an der Schnittstelle von Mathematik und Informatik weiter.

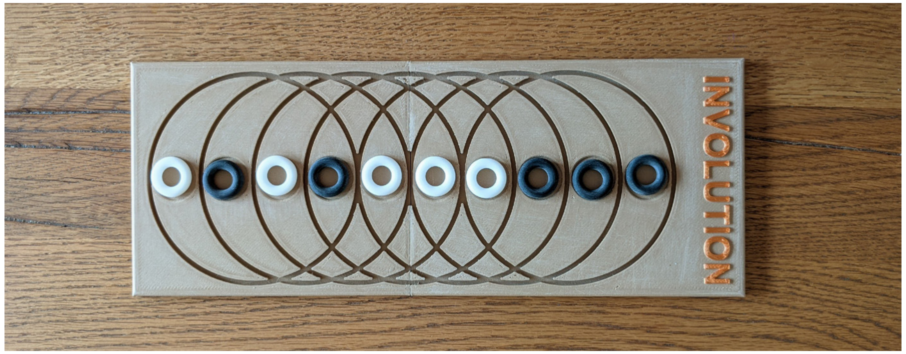
c. Didaktische Reduktion

Involution© bietet mehrere Challenges mit unterschiedlichem Schwierigkeitsgrad aus dem Bereich der „Shortest Path“-Problematik und stellt sie in den Rahmen des kooperativen und problemlösenden Lernens . Das Spiel regt einen gezielten kognitiven Prozess an, der anhand des verteilten Unterrichtsmaterials aktiv entwickelt werden kann. Bei der Planung und Durchführung der Unterrichtseinheit kann zugunsten eines freien Experimentierens mit Involution© und den zur Verfügung gestellten Anweisungen auf eine strikte Zeiteinteilung völlig verzichtet werden. Die Festigung von Wissen und Kompetenzen erfolgt durch das Lernen im Unterricht, indem die Schüler*innen der Klasse ihre Lösungen sowie die Gründe für ihre Entscheidung präsentieren.

05 | Methodische Analyse

In diesem Modul spielen die Schüler*innen ein Spiel namens Involution©. Das Spiel besteht aus einem Spielbrett und einer Kurbel. Das Spielbrett hat 10 Vertiefungen, die jeweils einen schwarzen oder weißen Ring enthalten.

Eine kreisförmige Kurbel wird auf das Spielbrett gesetzt und durch eine Kurbelbewegung werden die Positionen der Ringe verändert. Ziel des Spiels ist es, die Position der Ringe so zu verändern, dass man ausgehend von einer Startkonfiguration zu einer bestimmten Zielkonfiguration gelangt.



Involution 1

Das Spiel

Die Lehrkraft erklärt kurz das Prinzip des Spiels. Der Einstieg in die Thematik erfolgt über das Spiel selbst. Die Schüler*innen bilden Zweiergruppen. Jede Gruppe erhält eine Box, die ein Spielbrett (zum Zusammenbauen) und Aufgabenkarten enthält. Auf jeder Karte sind zwei Konfigurationen abgebildet. Die Aufgabe besteht darin, durch Drehen der Kurbel von der linken zur rechten Konfiguration zu wechseln. Die Schüler*innen beginnen, die Aufgaben in der vorgegebenen Reihenfolge zu lösen. Diese Aktivität ist die erste von vielen Challenges in diesem Modul. Alle Challenges sind in den Materialien beschrieben.

Nach diesem Einstieg in die Thematik gehen die Schüler*innen zu den verschiedenen Challenges über. Im besten Fall werden alle Challenges gelöst. Solche Challenges, die mit einem Sternchen gekennzeichnet sind, sind schwieriger (und mathematischer). Sie sind für motivierte Schüler*innen gedacht. Es gibt auch optionale Challenges (wie im Stundenverlaufsplan (siehe unten) angegeben), das Modul funktioniert jedoch auch ohne diese Challenges.

Challenge 2: Nach der ersten Phase, sobald die Schüler*innen das Prinzip des Spiels verstanden haben, gehen sie zu Challenge 2 über. Die Lehrkraft teilt die Klasse in drei Gruppen auf; eine Gruppe bearbeitet die erste Unterchallenge, die andere die zweite und die andere die dritte. Hinweis: Die erste Challenge ist etwas einfacher als die beiden anderen. Dies ist bei der Gruppenaufteilung und der Verteilung der Challenges zu berücksichtigen.

Sobald die Challenge gelöst ist, müssen sich die Schüler*innen in ihrer Gruppe abstimmen und dann ihr Ergebnis dem Rest der Klasse mitteilen.

Challenge 3*: Die Challenge 3 ist schwieriger und optional. Sie ist für leistungsstarke und motivierte Schüler*innen gedacht.

Wie gelangt man zu allen Konfigurationen?

Dieser Abschnitt hilft den Schüler*innen bei der Lösung von Challenge 3. Er richtet sich daher nur an diejenigen Schüler*innen, die sich mit der Challenge 3 beschäftigt haben. Von den in diesem Abschnitt vorgestellten Challenges ist die Challenge 9 besonders schwer. Wir überlassen es der Lehrkraft zu entscheiden, welche Schüler*innen diesen Teil mit oder ohne Challenge 9 bearbeiten können/sollen.

Dieser Abschnitt führt die Schüler*innen über das mathematische Prinzip der Induktion zu einer Lösung von Challenge 3 (ohne dieses Prinzip explizit zu erwähnen oder auszuführen).

Eine optimale Lösung

Challenge 10: Bei Challenge 10 suchen die Schüler*innen nach einer Lösung und zählen die Anzahl der Kurbelbewegungen bis zur Lösung. Sie vergleichen ihre Lösungen untereinander.

In einer moderierten Diskussion kommen die Schüler*innen zu dem Schluss, dass eine Lösung besser ist als eine andere, wenn sie weniger Kurbelbewegungen erfordert.

Im Klassenplenum wird gemeinsam eine Definition ähnlich wie die folgende erarbeitet: Eine Lösung ist optimal, wenn sie mit möglichst wenigen Kurbelbewegungen gefunden wird.

Auf der Suche nach einer optimalen Lösung

Wie findet man eine solche optimale Lösung? Kontextualisieren Sie diese Frage mit der Suche nach einer möglichst kurzen Strecke im New-Yorker U-Bahn-Netz, die im Kurs *Digital Sciences* 1 behandelt wird.[1]

Wie kann man das Spiel in Form eines Graphen darstellen?

Dieser Abschnitt beginnt mit einem ganz einfachen Beispiel: das Spiel Involution© wird vorgegeben, aber mit nur fünf Ringen und einem verkleinerten Spielbrett. Die fünf Vertiefungen ganz rechts dürfen nicht benutzt werden. Es gibt also nur zwei mögliche Positionen, an denen die Kurbel angesetzt werden kann.

Challenge 11: Die Lehrkraft teilt die Klasse in drei Gruppen ein. Innerhalb der Gruppen spielen die Schüler*innen das Spiel weiter in Zweiergruppen.

- Die erste Gruppe spielt mit 1 schwarzen Ring (und 4 weißen Ringen).
- Die zweite Gruppe spielt mit 2 schwarzen Ringen (und 3 weißen Ringen).
- Die dritte Gruppe spielt mit 3 schwarzen Ringen (und 2 weißen Ringen).

Die Schüler*innen lösen ihre jeweilige Aufgabe in Zweiergruppen. Anschließend stimmen sie sich in ihrer Gruppe ab und teilen ihr Ergebnis dem Rest der Klasse mit.

Challenge 12*: Diese Challenge ist schwieriger und optional.

Die Schüler*innen gehen dazu über, das Spiel grafisch darzustellen (mit dem verkleinerten Spielbrett).

Challenge 13: Für dieses, sowie auch die folgenden Challenges, muss das Konzept von Graphen eingeführt, respektiv kurz wiederholt werden. Die Schüler*innen arbeiten in Zweiergruppen. Anschließend vergleichen sie ihr Ergebnis. Sie werden feststellen, dass der zweite Fall und der dritte Fall genau gleich sind. In einer moderierten Diskussion werden die Schüler*innen zu der Erkenntnis hingeführt, dass diese beiden Fälle symmetrisch sind.

Challenge 14: Sobald der Graph erstellt ist, wird die Challenge 12 viel einfacher. An den Graphen können die Schüler*innen die Antwort deutlich ablesen.

In der nächsten Phase gehen die Schüler*innen wieder zum Spiel mit 10 Ringen über (5 schwarze und 5 weiße Ringe).

Challenge 15*: Diese Challenge ist optional. Sie soll den Schüler*innen vor Augen führen, dass das Zählen umso komplizierter wird, je komplexer das Problem ist. Viele Schüler*innen sind sich dessen nicht bewusst, da das Zählen bei wenigen Objekten sehr einfach ist.

Die grafische Darstellung des Spiels Involution© ist gegeben, aber sie ist zu groß, um mit bloßem Auge den kürzesten Pfad zu finden.

Auf der Suche nach einem Algorithmus

Um die Suche nach einem Algorithmus zu erleichtern, gehen die Schüler*innen zu 6 Ringen über (3 schwarze und 3 weiße). Der Graph bei Involution© mit 6 Ringen ist klein genug, um manuell den kürzesten Pfad zu finden (im Gegensatz zum Graph bei Involution© mit 10 Ringen), aber groß genug, damit die Lösung nicht zu leicht ist (im Gegensatz zu Involution© mit 5 Ringen).

Challenge 16 und 17: Diese Challenges dienen dazu, die Schüler*innen an einen Algorithmus heranzuführen. Sie werden in Zweiergruppen bearbeitet. Die Ergebnisse werden dann in der Klasse in Form einer moderierten Diskussion besprochen.

Challenge 18: Bei dieser Challenge wird der Algorithmus manuell erstellt. Die Schüler*innen sollen informelle Anweisungen verfassen. Punkt b) ist komplexer. Er ist nur für motivierte Schüler*innen gedacht.

Beim Zeichnen des Ablaufdiagramms können die Schüler*innen entweder alles selbst zeichnen oder die Arbeitshilfe verwenden, in der die Felder bereits vorgegeben sind. Anschließend sollen die Schüler*innen die Felder in die richtige Reihenfolge bringen und die Pfeile hinzufügen, um ein Ablaufdiagramm zu gestalten.

Challenge 19: Diese Challenge dient nur dazu, den Schüler*innen vor Augen zu führen, dass der allgemeine Algorithmus genau der ist, den sie gerade erstellt haben. Diese Challenge sollte direkt in einer moderierten Diskussion bearbeitet werden.

Challenge 20: In der moderierten Diskussion sollen die Schüler*innen erkennen, dass der Algorithmus, den sie gerade erstellt haben, ein Algorithmus des kürzesten Pfads darstellt. Diese Arten von Algorithmen werden auch von GPS-Navigationssystemen verwendet (siehe 1.6.02 Die Bedeutung von Graphen in den Digitalwissenschaften).

[1] Digital Sciences 7ᵉ Themenachse 1

06 Differenzierungsmöglichkeiten

Für Anfänger

Das Modul ist durchaus machbar, wenn man die Challenges und den Abschnitt weglässt, die mit einem Sternchen markiert sind. Ermutigen Sie schwache Schüler*innen, das Modul ohne diese Challenges zu absolvieren.

Für besonders motivierte Schüler*innen

Beziehen Sie bei besonders motivierten Schüler*innen die mit einem Sternchen versehenen Challenges (und den mit einem Sternchen markierten Abschnitt) mit ein. Diese Challenges sind mathematischer und erfordern logisches Denken. Es gibt eine Challenge, die mit zwei Sternchen gekennzeichnet ist. Diese ist nur für sehr leistungsstarke Schüler*innen geeignet, die mit dem Thema gut zurechtkommen.

Bei den Challenges 2 und 11 kann die Lehrkraft die drei Gruppen so bilden, dass die Gruppe, die den etwas einfacheren Fall bearbeitet, aus leistungsschwächeren Schüler*innen besteht.

Die Challenge 18, die Haupt-Challenge, bei der der Algorithmus erstellt wird,
hat drei Schwierigkeitsstufen:

1. Erstellung eines Algorithmus, der nur die geringstmögliche Anzahl von Kurbelbewegungen zählt und/oder bei dem die Felder des Ablaufdiagramms bereits vorgegeben sind.
2. Dasselbe ohne die Felder des Ablaufdiagramms
3. Erstellung eines Algorithmus, der die geringstmögliche Anzahl von Kurbelbewegungen und die Folge der Bewegungen zählt

07 Weitere im Rahmen der Unterrichtsreihe zu erfüllende Qualitätskriterien

a. **Luxemburgischer Kontext:** Das Spiel Involution© wurde von Hugo Parlier und Bruno Teheux, zwei Mathematikwissenschaftlern der Universität Luxemburg, entwickelt. In seiner aktuellen Form kann es an die Inhalte des luxemburgischen Medienkompasses anknüpfen und entspricht der Themenachse I des Fachs *Digital Sciences*.
b. **Differenzierung:** In diesem Modul sind einige Challenges optional. Die mit einem Sternchen markierten Challenges sind schwieriger als die anderen und können je nach Niveau der Klasse und/oder der Schüler*innen bearbeitet oder weggelassen werden.
c. **Medienkompass:** Vgl. die angestrebten Lernziele des Medienkompetenzrahmens innerhalb der didaktischen Analyse des vorliegenden Dokuments.
d. **4K-Modell: Kommunikation, Kollaboration, Kreativität, kritisches Denken:** Das 4K-Modell wird durch die verschiedenen Sozialformen und Unterrichtsaktivitäten auf unterschiedliche Weise aufgegriffen.
e. **Bezug zur aktuellen Forschung:** Die Challenges von Involution© gehören zur Familie der Rekonfigurationsproblematik (Combinatorial reconfiguration), einem aktuellen Forschungsgebiet, das in der Mathematik und Informatik untersucht wird. Die Algorithmen des kürzesten Pfads sind zwar schon über 50 Jahre alt, aber immer noch ein aktives Thema in der Forschung, um Anwendungen für die Suche nach dem kürzesten Pfad zu optimieren.
f. **Bezug zur Forschung in Luxemburg:** Das Spiel Involution wurde von zwei Wissenschaftlern der Universität Luxemburg entwickelt. In dem Podcast in 1.7 Wissenschaftler*innen kommen zu Wort erklärt Hugo Parlier, wie er Spiele entwickelt, in die die Ergebnisse seiner Forschung einfließen.

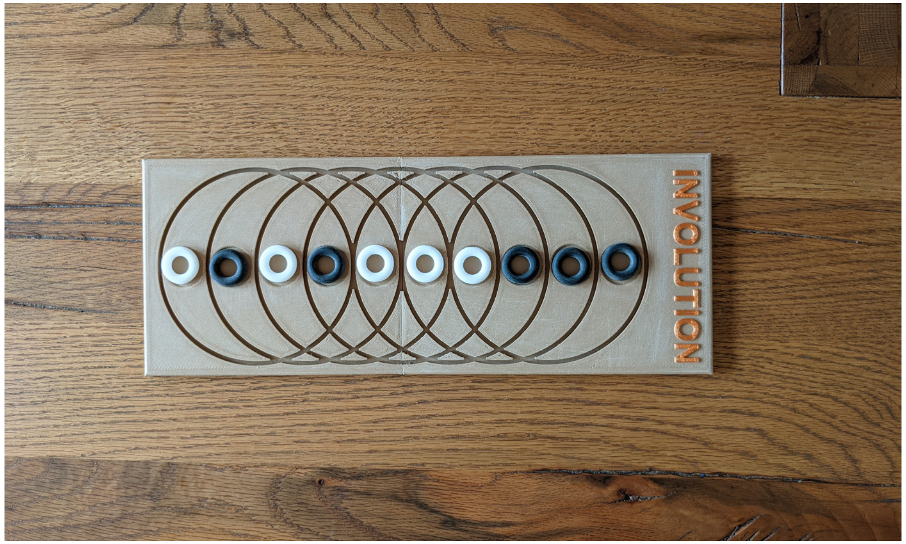
Referenzen:
Institut national de recherche en sciences et technologies du numérique (INRIA). (2020). *Éducation et Numérique : enjeux et défis.* Livre Blanc N 04. https://hal.inria.fr/hal-03051329v2/document
Han, Sheon. (2022). How to Write Software With Mathematical Perfection. *Quantamagazine.* https://www.quantamagazine.org/computing-expert-says-programmers-need-more-math-20220517

Thema der Sitzung: Involution					
Lernziele und in der Sitzung zu entwickelnde Kompetenzen: Die Schüler*innen kennen die Grundlagen des algorithmischen Problemlösens und haben diese durch Anwendung erprobt. Dabei haben sie kooperativ Lösungsstrategien entwickelt und diese auf der Grundlage einer eigenständigen Dokumentation metakognitiv reflektiert.					
OPTIONALE und FAKULTATIVE Challenges: In der folgenden Übersicht wird nach optionalen und fakultativen Aufgaben unterschieden. Je nach Geschwindigkeit, mit der die einzelnen Phasen von den SuS bearbeitet werden, kann also spontan entschieden werden, ob einzelne Aufgabenstellungen ausgelassen oder hinzugefügt werden sollen.					
Evaluation (falls geplant): In 1.6 Evaluationsmöglichkeiten werden mehrere Bewertungsmöglichkeiten mit unterschiedlichen Schwierigkeitsgraden vorgeschlagen.					
Zeit	**Phasen**	**Inhaltliche Schwerpunkte**	**Sozialform / Methoden**	**Materialien und Medien**	**Lernprozess**
10min	Einstieg	**Erlernen der Funktionsweise des Rätsel-Spiels Involution© Challenge 1**	• Tandems • Lehrer*innenvortrag	• Involution© • Arbeitsblätter mit Anleitung und Aufgaben (in gedruckter Form oder auf Tablet)	Die SuS ... **verstehen** das Konzept und die Funktionsweise des Rätsel-Spiels Involution.
20min	Involution©	**[OPTIONAL] Erarbeitung Challenges 2 & 3**	• Tandems	• Involution© • Arbeitsblätter mit Anleitung und Aufgaben (in gedruckter Form oder auf Tablet)	Die SuS ... **erarbeiten** gemeinsam eine Problemlösung für die Aufgabenstellung. ... **reflektieren** ihre Strategien und Lösungswege.
20min	Involution© Eine optimale Lösung	**Erarbeitung Challenge 10**	• Tandems	• Involution© • Arbeitsblätter mit Anleitung und Aufgaben (in gedruckter Form oder auf Tablet)	Die SuS ... **erarbeiten** gemeinsam eine Problemlösung für die Aufgabenstellung. ... **reflektieren** ihre Strategien und Lösungswege.
25min	Involution© Auf der Suche nach einer optimalen Lösung	**Erarbeitung Challenges 11-15 [OPTIONAL: Challenges 12, 14, 15]**	• Tandems	• Involution© • Arbeitsblätter mit Anleitung und Aufgaben (in gedruckter Form oder auf Tablet)	Die SuS ... **erarbeiten** gemeinsam eine Problemlösung für die Aufgabenstellung. ... **reflektieren** ihre Strategien und Lösungswege.
25min	Involution© Auf der Suche nach einem Algorithmus	**Erarbeitung Challenges 16-20**	• Tandems	• Involution© • Arbeitsblätter mit Anleitung und Aufgaben (in gedruckter Form oder auf Tablet)	Die SuS ... **erarbeiten** gemeinsam eine Problemlösung für die Aufgabenstellung. ... **reflektieren** ihre Strategien und Lösungswege.

1.3 Anweisungen für die Schüler*innen

01 | Das Spiel

In diesem Modul werden wir ein Spiel namens Involution© spielen. Das Spiel besteht aus einem Spielbrett und einer Kurbel. Das Spielbrett hat 10 Vertiefungen, die jeweils 1 Ring (einer bestimmten Farbe) enthalten.



Die Ringe können — wie im folgenden Video gezeigt — bewegt werden:

Involution 1

Challenge 1:

Bildet Zweiergruppen und spielt das Spiel Involution©: Versucht, die Challenges auf den Karten zu lösen.

Challenge 2:

Ist es möglich, ausgehend von der linken Konfiguration zur rechten Konfiguration zu gelangen?

○○○○○●●●●●	●○●○●○●○●○
○○○○○●●●●●	●●●●○●○○○○
○○○○○●●●●●	●●●○●●○○○○

Wenn ja, erklärt eure Kurbelbewegungen. Wenn nein, warum nicht?

Challenge 3*:

Kann man zu allen Ringkonfigurationen gelangen? Wenn ja, warum? Wenn nein, welche sind nicht möglich? *Begründet* eure Entscheidung.

02 | Wie gelangt man zu allen Konfigurationen?

Um klarer zu sehen, spielen wir das Spiel jetzt mit 9 weißen Ringen und einem schwarzen Ring.

Die entscheidende Frage ist folgende:

Challenge 4*:

Ist es möglich, von einer beliebigen Konfiguration zu folgender Konfiguration zu gelangen? *Begründet* eure Entscheidung.

●○○○○○○○○○

Spielen wir jetzt mit zwei schwarzen Ringen.

Challenge 5*:

Ist es möglich, von einer beliebigen Konfiguration zu folgender Konfiguration zu gelangen? Verwendet eure Lösung aus Challenge 4 und *begründet* eure Entscheidung.

●●○○○○○○○○

Spielen wir jetzt mit drei schwarzen Ringen.

Challenge 6*:

Verwendet die Lösung aus Challenge 5, um herauszufinden, ob ihr von jeder beliebigen Konfiguration zu folgender Konfiguration gelangen könnt.

●●●○○○○○○○

Spielen wir jetzt mit vier schwarzen Ringen.

Challenge 7*:

Verwendet die Lösung aus Challenge 6, um herauszufinden, ob ihr von jeder beliebigen Konfiguration zu folgender Konfiguration gelangen könnt.

●●●●○○○○○○

Challenge 8*:

Schaut euch nochmal Challenge 3* an.

Challenge 9:**

Und wenn wir das gleiche Spiel mit 12 Ringen spielen (6 schwarze und 6 weiße), ist es dann möglich, zu allen Konfigurationen zu gelangen? Und mit 100 Ringen (50 schwarze und 50 weiße)?

03 | Eine optimale Lösung

Challenge 10:

Findet ausgehend von der folgenden Konfiguration

○○○○○●●●●●

die folgende Konfiguration

●●●●●○○○○○

und zählt die Anzahl der Kurbelbewegungen.

Vergleicht eure Lösungen untereinander. Welche ist die beste? *Legt ein Kriterium fest*, nach dem beurteilt werden kann, welche Lösung besser ist als eine andere.

Vervollständigt nach der Diskussion die folgende Definition:

Eine Lösung ist optimal, wenn ____.

04 | Auf der Suche nach einer optimalen Lösung

Wir werden unser Spiel nun bildhaft darstellen, um nach einer optimalen Lösung zu suchen. Erinnern wir uns daran, dass im Fach *Digital Sciences 1*[1] einer der ersten Algorithmen, mit denen wir zu tun hatten, ein Algorithmus war, um mit der New Yorker U-Bahn eine Route zu finden. So wie der U-Bahn-Plan eine grafische Darstellung des echten Streckennetzes ist, werden wir nun das Spiel grafisch darstellen: Die Knoten des U-Bahn-Netzes sind die verschiedenen Konfigurationen des Spiels (bei der U-Bahn waren die Knoten die Stationen) und zwei Konfigurationen sind durch eine Linie verbunden, wenn man mit einer Kurbelbewegung von einer Konfiguration zur anderen gelangen kann (im Fall der U-Bahn waren zwei Stationen durch eine Linie verbunden, wenn man mit der U-Bahn direkt von einer Station zur anderen fahren konnte).

Schauen wir uns das anhand eines Beispiels an. Wir spielen das Spiel Involution©, aber nur mit 5 Ringen und einem verkleinerten Spielbrett (nehmt dasselbe Spielbrett, aber ihr dürft die 5 Vertiefungen ganz rechts nicht benutzen). Daher dürft ihr nur zwei Bewegungen mit der Kurbel machen.

Challenge 11:

Die Klasse wird in 3 Gruppen geteilt.

- Die erste Gruppe spielt mit 1 schwarzen Ring (und 4 weißen Ringen).
- Die zweite Gruppe spielt mit 2 schwarzen Ringen (und 3 weißen Ringen).
- Die dritte Gruppe spielt mit 3 schwarzen Ringen (und 2 weißen Ringen).

Zeichnet in Zweiergruppen alle möglichen Konfigurationen eures Involution©-Spezialfalls.

Diskutiert und *überlegt* gemeinsam, wie viele Konfigurationen möglich sind.
Präsentiert anschließend eure Lösung und eure Erklärung vor der ganzen Klasse.

Challenge 12*:

Findet heraus, ob ihr in eurem Fall zu jeder beliebigen Konfiguration gelangt, und *begründet* eure Entscheidung.

Wir gehen jetzt zur grafischen Darstellung über.

Challenge 13:

Gebt jeder Konfiguration einen Namen (z. B. C_1, C_2, C_3 usw.). Zeichnet für jede gefundene Konfiguration aus Challenge 11 einen Knoten und notiert daneben dessen Namen. Verbindet dann zwei Knoten mit einer Linie, wenn man mit einer einzigen Kurbelbewegung von einer Konfiguration zur anderen gelangt. Vergleicht eure Graphen.

[1]Digital Sciences 7^{e} Themenachse 1

Challenge 14:

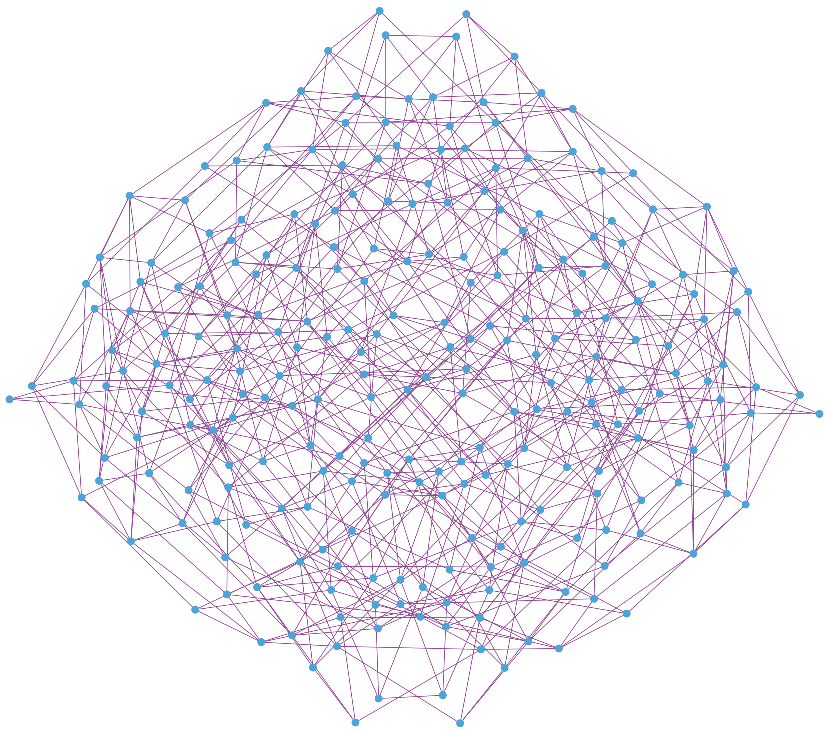
Geht zurück zu Challenge 12 (verwendet den Graphen, den ihr gerade erstellt habt).

Wir spielen jetzt wieder mit 10 Ringen.

Challenge 15*:

Könnt ihr die Anzahl der verschiedenen Konfigurationen berechnen oder zumindest eine Obergrenze angeben?

Insgesamt gibt es 252 verschiedene Konfigurationen. Das ergibt also einen Graphen mit 252 Knoten, der wie folgt aussieht.



Der Involution©-Graph mit 10 Ringen ist viel zu groß, um den kürzesten Pfad mit bloßem Auge zu sehen. Deshalb brauchen wir einen Algorithmus.

05 | Auf der Suche nach einem Algorithmus

Challenge 16:

Spielt das Spiel jetzt mit 6 Ringen (3 schwarze und 3 weiße) und einem verkleinerten Spielbrett (nehmt dasselbe Spielbrett, aber ihr dürft die 4 Vertiefungen ganz rechts nicht benutzen). Findet alle möglichen Konfigurationen und zeichnet den Graphen.

Challenge 17:

Verwendet den in Challenge 16 gefundenen Graphen, um zu bestimmen, was eine optimale Lösung ist, um zu der folgenden Konfiguration zu gelangen

○●○●○●

ausgehend von folgender Konfiguration:

○○○●●●

The Big Bang Theory Sheldon's Friendship Alogrithm

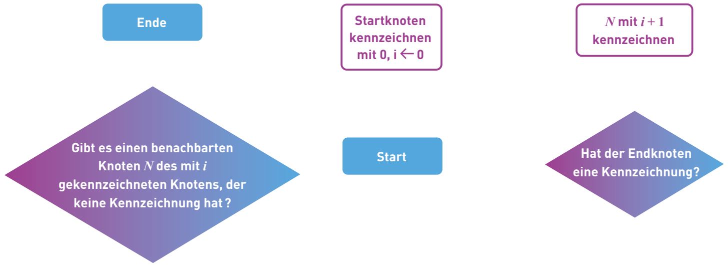
Challenge 18:

Lasst euch von der letzten Challenge inspirieren und sucht einen Algorithmus für

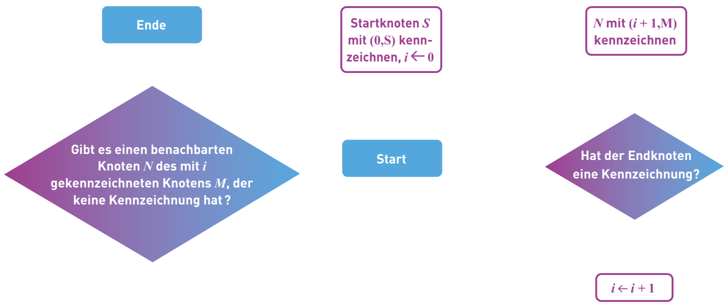
1. die geringstmögliche Anzahl an Kurbelbewegungen, um von einer Konfiguration zur anderen zu gelangen (sobald der Graph des Spiels erstellt wurde),
2. eine optimale Lösung, um von einer Konfiguration zur anderen zu gelangen (sobald der Graph des Spiels erstellt wurde).

Schreibt die informellen Anweisungen für euren Algorithmus auf und zeichnet ein Ablaufdiagramm.

Hilfe 1:



Hilfe 2:

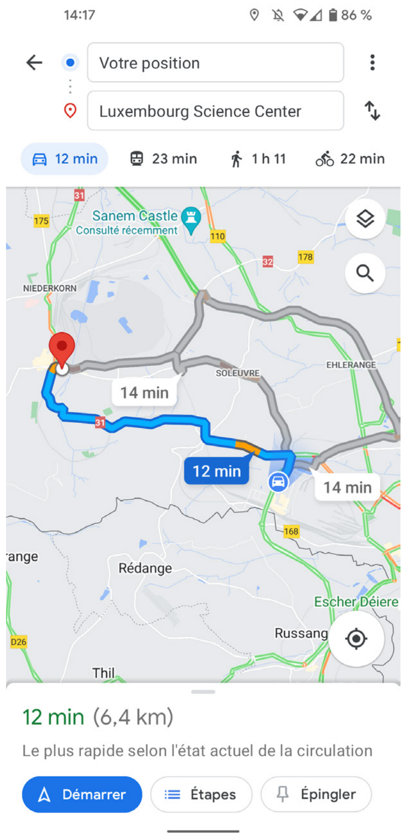


Challenge 19:

Verallgemeinert den Algorithmus im Spiel Involution© mit 10 Ringen.

Challenge 20:

Schaut euch das folgende Bild an:



Um welche Anwendung handelt es sich?

Sucht eine Verbindung zwischen dieser Anwendung und der Suche nach einer optimalen Lösung im Spiel Involution©.

1.4 Fächerübergreifende Ideen

Mathematik

Dieses Modul zielt exakt auf die vier Kompetenzen in Bezug auf mathematische Prozesse ab, die in den vom Bildungsministerium herausgegebenen Dokumenten *Compétences disciplinaires attendues à la fin de la classe de 6ᵉ* et *à la fin de la classe de 4ᵉ* und *Compétences disciplinaires attendues à la fin de la classe de 7G-6G-5G* aufgeführt sind. Dabei handelt es sich um folgende:

1. Problemlösung: Wie im Dokument zu den fächerspezifischen Kompetenzen erläutert, ist das Lösen mathematischer Probleme „einerseits durch den Einsatz allgemeiner Strategien und andererseits durch den Einsatz spezifischer Strategien gekennzeichnet". Genau das sollen die Schüler*innen in diesem Modul tun, indem sie zunächst das Spiel spielen und einige Beispiele ausprobieren und dann der von dem Modul vorgegebenen Strategie folgen. Dieses Modul ist ein gutes Beispiel dafür, dass die Problemlösung geschult wird, indem „aktiv an Problemen gearbeitet und über Methoden und Strategien zur Problemlösung nachgedacht wird".
2. Modellieren: In diesem Modul sollen die Schüler*innen zunächst das Problem vereinfachen und es dann in Form eines Graphen modellieren. Dieser Prozess entspricht exakt der Beschreibung der Kompetenz „Modellieren": „Es geht darum, die reale Situation zunächst zu vereinfachen und dann zu mathematisieren, d. h. sie mit mathematischen Werkzeugen zu beschreiben".
3. Argumentieren: Diese Kompetenz wird folgendermaßen beschrieben: „Mathematisches Argumentieren beginnt mit der Erkundung von Situationen, der Suche nach Strukturen und Beziehungen und der Formulierung von Vermutungen über mathematische Zusammenhänge". Genau das geschieht in den Challenges 3 bis 8 in den Materialien. In der ersten Übung werden die Schüler*innen aufgefordert, eine Vermutung zu formulieren, in den folgenden Übungen sollen sie ihre Vermutung dann beweisen.
4. Kommunizieren: Die Übungen in diesem Modul sind keine typischen Übungen aus dem Mathematikunterricht, sondern erfordern Begründung und Argumentation, um den Gedankengang zu erklären. Die Schüler*innen sollen also „mathematische Inhalte mithilfe der Alltagssprache und der mathematischen Sprache in geeigneter Weise darstellen".

Konkret wird mit diesem Modul eine Kompetenz aus dem Kapitel „Literalrechnungen" des neuen vorläufigen Mathematiklehrplans für die Klasse 6C erarbeitet. Diese Kompetenz wird wie folgt beschrieben: einen literalen Ausdruck entschlüsseln und in eine geordnete Reihe von Rechenanweisungen bringen und diesen Algorithmus formalisieren können.

Darüber hinaus kann das Spiel Involution© verwendet werden, um die Konzepte der zentralen Symmetrie und der Rotationssymmetrie zu veranschaulichen. Die Kurbelbewegung im Spiel ist nichts anderes als eine zentrale Symmetrie oder Rotation um 180 Grad. Diese geometrischen Transformationen werden in der 5ᵉ und 4ᵉ behandelt und bereiten den Schüler*innen oft Probleme: Die Achsensymmetrie erscheint ihnen viel eingängiger und sie haben Schwierigkeiten, diese neuen Transformationen zu visualisieren. Involution© ist eine andere Möglichkeit der Visualisierung.

In der Unterstufe des allgemeinen Sekundarunterrichts beginnen die Schüler*innen bereits, sich mit Wahrscheinlichkeiten vertraut zu machen. Involution© ermöglicht das Zählen und Abzählen (wie viele verschiedene Konfigurationen gibt es?) und das Berechnen von Wahrscheinlichkeiten auf viele verschiedene Arten.

Zu guter Letzt ist dieses Modul eine Vorbereitung für das Beweisverfahren der Induktion. In den Challenges 4 bis 8 in den Materialien werden die Schüler*innen Schritt für Schritt dazu angeleitet, einen Induktionsbeweis durchzuführen (ohne dass dieser mathematische Begriff erwähnt wird).

Geografie

Über die Navigationsalgorithmen (wie GoogleMaps) kann eine Verbindung zu den Karten hergestellt werden, die von Navigationsanwendungen verwendet werden. Orientierung und Karten stehen auf dem Geografie-Lehrplan für die Klassen 7C und 7G.

1.5 Evaluationsmöglichkeiten

Die Gruppendiskussion über die Lösungsmethoden für die verschiedenen Challenges, die mit Involution© gelöst werden sollen, kann durch eine Evaluation ergänzt werden, die zur Festigung der erworbenen Kenntnisse und Kompetenzen beiträgt.

Erstellen eines Lösungshefts für Involution© mit 3 Farben

Spielen wir Involution© mit 3 Farben: 2 schwarze Ringe, 2 weiße Ringe und 1 Vertiefung ohne Ring (die Vertiefung ohne Ring stellt die dritte Farbe dar). Die Schüler*innen bekommen die Aufgabe, ein Heft mit optimalen Lösungen für eine (von der Lehrkraft) zu bestimmende Anzahl von Challenges zu erstellen. Die Wahl der Präsentationsform bleibt den Schüler*innen überlassen – es kann sich um einfache Prozessbeschreibungen, Zeichnungen, rein visuelle Darstellungen oder ein von den Schüler*innen selbst erstelltes Video handeln.

Um optimale Lösungen zu finden, sollen die Schüler*innen den entsprechenden Graphen für Involution© mit 3 Farben erstellen und den kürzesten Pfad in diesem Graphen suchen (Achtung: Diesen Pfad gibt es vielleicht nicht immer).

Zum Schluss können die Schüler*innen ihre fertigen Hefte untereinander tauschen und mithilfe des Spiels überprüfen, ob alles richtig ist. So können sich die Schüler*innen gegenseitig beurteilen und Feedback geben.

Es ist auch möglich, größere und komplexere Projekte daraus zu machen. Wir stellen hier drei vor: Das erste ist computerbasierter (und erfordert Programmierungen), das zweite ist eher im Bereich Sozialwissenschaften und Kreativität angesiedelt und das dritte richtet sich an Schüler*innen, die sehr mathematikaffin sind.

Involution© programmieren

Ziel ist es, ein Programm (in Scratch oder Python) zu schreiben und dafür den in den Challenges 15 und 16 erstellten Algorithmus zu verwenden, um das Spiel Involution© zu lösen.

Es gibt mehrere Schwierigkeitsgrade:

Ein Programm schreiben, das als Input zwei Konfigurationen des Spiels Involution© verwendet und als Output die geringstmögliche Anzahl von Kurbelbewegungen angibt, um von einer Konfiguration zur anderen zu gelangen.

Ein Programm schreiben, das als Input eine Konfiguration des Spiels Involution© verwendet und als Output eine Folge von Kurbelbewegungen angibt (nicht notwendigerweise die Folge, die der optimalen Lösung entspricht), um von dieser Konfiguration zu der Konfiguration zu gelangen, bei der sich alle weißen Ringe auf der einen Seite und alle schwarzen Ringe auf der anderen Seite befinden.

Ein Programm schreiben, das als Input eine Konfiguration des Spiels Involution© verwendet und als Output die optimale Folge von Kurbelbewegungen angibt, um von dieser Konfiguration zu der Konfiguration zu gelangen, bei der sich alle weißen Ringe auf der einen Seite und alle schwarzen Ringe auf der anderen Seite befinden.

Ein Spiel für zwei Spieler*innen erfinden

Die Schüler*innen sollen sich eine Version überlegen, mit der das Spiel Involution© mit zwei Spieler*innen gespielt werden kann. Sie sollen ein kooperatives Spiel und ein Wettbewerbsspiel erfinden. Anschließend haben sie die Aufgabe, klare und eindeutige Spielregeln aufzustellen. Welche Präsentationsform gewählt wird, bleibt den Schüler*innen überlassen (kleines Heft, Zeichnungen, digitales Medium usw.).

Und mit 3 Farben?

Spielen wir Involution© mit 3 Farben: n schwarze Ringe, n weiße Ringe und 1 Vertiefung ohne Ring (die Vertiefung ohne Ring stellt die dritte Farbe dar). Der Wert n kann einen beliebigen Wert größer oder gleich 2 annehmen. Die Frage, die die Schüler*innen lösen müssen, lautet: Von welchem Wert n ausgehend gelangt man von jeder beliebigen Konfiguration zu allen Konfigurationen?

1.6 Mehr zum Thema

01 | Die Algorithmen BFS und Dijkstra

Der Algorithmus, den die Schüler*innen in diesem Modul kennengelernt haben, ist ein Algorithmus für die Suche nach dem kürzesten Pfad. In der Fachsprache heißt dieser Algorithmus Breadth-first Search-Algorithmus oder einfach BFS-Algorithmus. Im Deutschen wird er auch als „Breitensuche" bezeichnet. Er wurde 1945 erfunden und ermöglicht es, einen Graphen zu durchlaufen, indem zunächst ein Ausgangsknoten, dann seine Folgeknoten, dann die nicht besuchten Folgeknoten der Folgeknoten usw. besucht werden. Mit dem Breitensuchalgorithmus können die Entfernungen aller Knoten von einem Ausgangsknoten in einem Graphen berechnet werden. Das Grundprinzip des BFS-Algorithmus besteht darin, den Startknoten an den Anfang zu setzen und dann alle seine Nachbarn, die noch nicht besucht wurden, zu einer Warteschlange hinzuzufügen. Bereits besuchte Knoten werden markiert, damit ein und derselbe Knoten nicht mehrmals besucht wird. Dieser Algorithmus funktioniert nur bei nicht gewichteten Graphen, d. h. Graphen, bei denen die Knoten durch Kanten verbunden sind, deren Kantenlängen alle gleich sind und die kein Kantengewicht haben. Das trifft auf die Graphen in diesem Modul zu: Zwei Knoten (hier: Konfigurationen) sind durch eine Kante verbunden, wenn man mit einer einzigen Kurbelbewegung von einer Konfiguration zur anderen gelangen kann. BFS dagegen funktioniert bei gerichteten und nicht gerichteten Graphen. Ein gerichteter Graph ist ein Graph, bei dem die Kanten zwischen zwei Knoten gerichtet sind, das heißt, sie verlaufen von einem Knoten zu einem anderen (Diestel, 2018) bzw. sie können nur in eine Richtung durchlaufen werden. Da das Spiel Involution© symmetrisch ist (wenn man von einer Konfiguration A zu einer Konfiguration B mit nur einer Kurbelbewegung gelangen kann, kann man mit der entgegengesetzten Bewegung von B nach A gelangen, was exakt die gleiche Bewegung ist), sind die Graphen in diesem Modul nicht gerichtet. Die Fakultät für Computerwissenschaften der Cornell University hat Videos mit einer Einführung in den BFS-Algorithmus veröffentlicht, mit einer Dauer von insgesamt 15 Minuten (Gries, n.d.).

In anderen Situationen als einem Spiel müssen die Graphen gewichtet sein. Der Graph des Straßennetzes, den wir in diesem Modul kennengelernt haben, ist ein gewichteter Graph: Die Entfernung zwischen zwei Kreuzungen variiert situationsabhängig und der Graph muss dem Rechnung tragen, indem er die Kanten gewichtet. Der kürzeste Pfad ist also nicht mehr unbedingt der Pfad, der die wenigsten Knoten durchläuft. BFS funktioniert daher nicht bei gewichteten Graphen. Der Algorithmus von Dijkstra ist ein bekannter Algorithmus, der den kürzesten Pfad für gewichtete (oder ungewichtete) gerichtete oder nicht gerichtete Graphen berechnet. Er ist nach seinem Erfinder Edsger Dijkstra benannt und wurde erstmals 1959 veröffentlicht (Dijkstra, 1959). Der Algorithmus von Dijkstra funktioniert wie folgt: Der Algorithmus beginnt bei einem Startknoten und berechnet die Entfernungen von diesem Knoten zu allen anderen Knoten im Graphen. Anschließend wählt er den Knoten mit der geringsten Entfernung zum Startknoten aus. Dieser Knoten wird zusammen mit dem Startknoten einer Warteschlange hinzugefügt, und anschließend werden alle Entfernungen neu bewertet: Wenn die Entfernung vom Startknoten zu einem Knoten kleiner ist, wenn ein neuer Knoten durchlaufen wird, wird sie durch die geringere Entfernung ersetzt. Der Algorithmus fährt auf diese Weise fort, bis er alle Knoten durchlaufen hat. So werden die Mindestentfernungen vom Startknoten zu allen anderen Knoten ermittelt. Die Fakultät für Computerwissenschaften der Cornell University hat auch eine Einführung in den Algorithmus von Dijkstra veröffentlicht (Gries, n.d.).

02 | Die Bedeutung der Graphentheorie in den Digitalwissenschaften

Der Mathematiker J. H. C. Whitehead sagte immer wieder, dass „die Kombinatorik das Armenhaus der Topologie ist" (Cameron, 2011). Die Kombinatorik ist eine mathematische Disziplin, zu der auch die Graphentheorie gehört. Diese Einstellung zur Graphentheorie war zu Beginn des 20. Jahrhunderts unter vielen Mathematikern weit verbreitet, hat sich aber im Laufe der Zeit stark oder sogar vollständig gewandelt. Die Graphentheorie ist ein wichtiges Werkzeug zur Untersuchung von Verbindungen in komplexen Systemen. Sobald ein System schematisch in Form von Knoten und Kanten dargestellt werden kann, findet die Graphentheorie Anwendung. Diese Systeme können sehr unterschiedlich sein: Straßennetze, städtische Netze mit Computerdaten ... Es gibt in der heutigen digitalen Welt unzählige Anwendungen. Hier einige sehr wichtige Beispiele (Flovik, 2020):

- Ansteckung mit dem Covid-19-Virus in der Gemeinschaft
- Ranking von Websites in Suchmaschinen (wie Google)
- Netzsicherheit
- GPS-Navigationssysteme
- usw.

Nehmen wir als Beispiel die GPS-Navigationssysteme (wie GoogleMaps, Waze usw.), die auch in diesem Modul vorkommen. GPS-Navigationssysteme gehören zu den Anwendungen der Graphentheorie, die direkt mit unserem Alltag zu tun haben: Sehr oft überlassen wir es diesen Systemen, uns auf dem kürzesten (oder schnellsten) Weg von A nach B zu bringen. Ein Straßennetz kann durch einen Graphen dargestellt werden, dessen Knoten Kreuzungen und Gebäude sind und dessen Kanten die Straßenabschnitte sind, die sie verbinden. Da bestimmte Straßen nur in eine Richtung führen, ist der Graph gerichtet (im Gegensatz zum Involution©-Graphen, der nicht gerichtet ist). Das Navigationssystem berechnet einen optimalen Pfad zwischen zwei Punkten im Netz, basierend auf den von den Benutzer*innen festgelegten Kriterien (es optimiert entweder die Länge der Strecke, die Fahrzeit oder bevorzugt Autobahnen usw.). Stellen wir uns in diesem Zusammenhang vor, dass das System versucht, die Länge der Strecke zu minimieren. Um das Problem zu modellieren, müssen wir dem gerichteten Graphen, der das Straßennetz modelliert, Informationen hinzufügen. Man weist also jedem Straßenabschnitt ein Gewicht zu, das genau der Länge dieses Abschnitts entspricht. Navigationssysteme verwenden daher gerichtete und gewichtete Graphen. Der Algorithmus zur Suche nach dem kürzesten Pfad, den die Navigationssysteme verwenden, basiert auf dem Algorithmus von Dijkstra (Teheux, 2019).

Interessant ist, dass die GPS-Navigationssysteme den Dijkstra-Algorithmus invers verwenden. Konkret bedeutet das: Wenn das Ziel der Benutzer*innen darin besteht, von Knoten A zu Knoten B zu gelangen, dann wendet das Navigationssystem den Algorithmus von Dijkstra auf denermittelten Graphen an, indem es die Kanten des Straßennetzgraphen mit Knoten B als Ausgangsknoten ausgibt. Das Ergebnis ist dann der kürzeste Weg von B nach A. Man muss diesen Weg aber nur umkehren, dann erhält man den gewünschten Weg. Die Idee hinter diesem inversen Verfahren ist eine Arbeitsersparnis bei den Berechnungen. Es kommt nicht selten vor, dass wir als Autofahrer*innen irgendwann den vom Navi empfohlenen Weg verlassen (wir fahren falsch, eine Straße ist für den Verkehr gesperrt, wir ändern unseren Plan in letzter Minute usw.). Wenn der Algorithmus die Suche nach dem kürzesten Weg in der richtigen Richtung durchgeführt hätte, müsste er von vorne beginnen (da wir uns an einem Ort befinden, der in den Berechnungen des Algorithmus nicht vorgesehen war) und alle vorher vorgenommenen Berechnungen wären hinfällig. Wenn man dagegen am Ende beginnt, hat der Algorithmus bereits den kürzesten Weg von einem beliebigen Punkt (innerhalb eines angemessenen Radius) zum Endpunkt berechnet. Dadurch ist der Algorithmus in der Lage, uns auch bei einer plötzlichen Änderung der Route schnell eine Alternative anzubieten (Teheux, 2019).

Obwohl die reine Graphentheorie eine abstrakte und mathematische Disziplin ist, wird deutlich, dass sie viele konkrete und interessante Anwendungen in der Informatik hat. Wie sagte der international bekannte Informatiker Leslie Lamport so schön: „Mathematik ist ein Teil des Computerdenkens" (Han, 2022).

Referenzen
Cameron, Peter J. (2011). Aftermath. Preprint. https://arxiv.org/pdf/1111.4050.pdf
Diestel, Reinhard. (2018). *Graph theory (5th Ed.).* Graduate Texts in Mathematics, 173. Berlin : Springer.
Dijkstra, Edsger W. (1959). A note on two problems in connexion with graphs. *Numer. Math.*, 1, 269 – 271. https://doi.org/10.1007/BF01386390
Flovik, Vegard. (2020). What is Graph Theory, and why should you care. From graph theory to path optimization. *Towards Data Science.* https://towardsdatascience.com/what-is-graph-theory-and-why-should-you-care-28d6a715a5c2
Gries, David. (n.d.). *Depth-first search and breadth-first search.* https://www.cs.cornell.edu/courses/JavaAndDS/dfs/dfs01.html
Gries, David. (n.d.). *The shortest-path algorithm.* https://www.cs.cornell.edu/courses/JavaAndDS/shortestPath/shortestPath.html
Han, Sheon. (2022). How to Write Software With Mathematical Perfection. *Quantamagazine.* https://www.quantamagazine.org/computing-expert-says-programmers-need-more-math-20220517
Teheux, Bruno. (2019). À la recherche des chemins les plus courts. *Losange*, N 46, 45-54.

1.7 Wissenschaftler*innen kommen zu Wort: Dr. Hugo Parlier und Dr. Bruno Teheux

Hugo Parlier ist seit 2017 Professor im Fach Mathematik an der Universität Luxemburg. Seine Forschung umfasst verschiedene Themen aus den Bereichen Geometrie, Topologie und Kombinatorik. Er hat Mathematik an der École Polytechnique Fédérale de Lausanne studiert, wo er 2004 seinen Doktortitel erwarb. Nach Aufenthalten als Post-Doktorand in Madrid, Toronto und Genf wurde er mit Unterstützung des Schweizerischen Nationalfonds zum außerordentlichen Professor in Freiburg berufen.

Bruno Teheux ist ständiger Forscher an der Universität Luxemburg und forscht im Bereich der mathematischen Logik. Er promovierte 2009 an der Universität Lüttich in Belgien. Nachdem er zwei Jahre bei Animath in Paris tätig war, wechselte er 2012 an die Universität Luxemburg.

Die beiden Mathematiker möchten die breite Öffentlichkeit durch Aktivitäten an ihrer Leidenschaft für die Mathematik teilhaben lassen. So haben sie beispielsweise die Projekte *The Simplicity of Complexity* und Recreate: shapes from the collective Imagination ins Leben gerufen, die auf der Weltausstellung Dubai 2020 gezeigt wurden. Darüber hinaus wirken sie an dem Projekt *The Sound of Data* im Rahmen von Esch2022: Kulturhauptstadt Europas mit.

Neben ihren gemeinsamen Projekten war Bruno Teheux Herausgeber eines Pilotbandes mit Wissenschaftscomics, die von Doktorand*innen geschrieben wurden. Hugo Parlier ist Co-Autor eines interaktiven Mathematikbuchs für das Ipad namens *Mathema*. Dieses Buch enthält Rätsel für das iPad oder iPhone: Quadratis.

Interview mit Bruno Teheux und Ann Kiefer

Leider konnte Hugo Parlier nicht an unserem Interview teilnehmen. Daher verweisen wir auf den hervorragenden Podcast „Mäin Element", für den er interviewt wurde. Dieser Podcast wird vom Lëtzebuerger Journal in Zusammenarbeit mit dem Fonds National de la Recherche (FNR) produziert.

Podcast Mäin Element mit Hugo Parlier

ISSN 2738-9383

PROGRAMME FOR INNOVATIVE TEACHING AND TRAINING

PIT