Musteraufgaben Stufe 9-10

Hast du Interesse am Informatik Biber Wettbewerb teilzunehmen?

Ab dem 12.09.2022 kannst du am Schnupper-Biber teilnehmen.

Vom 7. November 2022 bis 18. November 2022 findet der richtige Biber-Wettbewerb statt. Du erreichst beide über diesen link:

https://wettbewerb.informatik-biber.de/

Deinen Benutzernamen und das Kennwort erhälst du von deiner Klassenleitung bzw. deinem Tutor (EF: DeutschlehrerIn / Q1 und Q2: LK-LehrerIn.

3-4: – 5-6: – 7-8: mittel 9-10: leicht 11-13: –

Blumenkasten

Peter liebt Blumen.

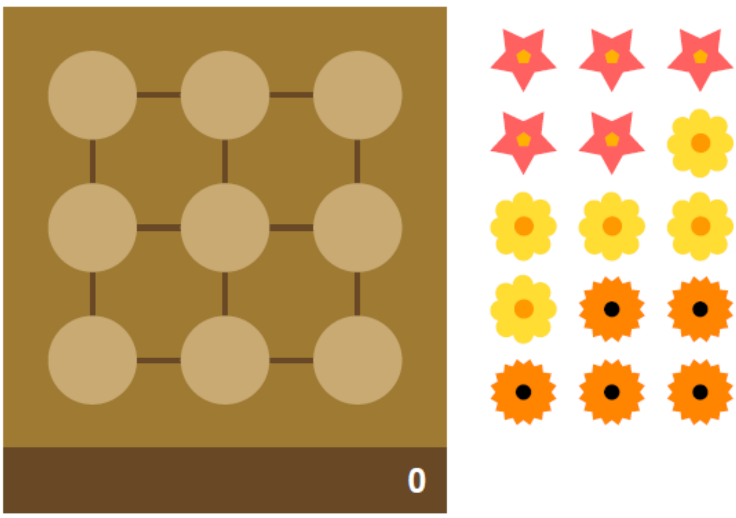
Er hat rote  , gelbe  und orange  Blumen.
Peter hat einen neuen Blumenkasten, mit Platz für 3x3 Blumen.
Dafür sucht er die perfekte Bepflanzung.

Um eine Bepflanzung zu bewerten, schaut Peter,
welche Blumenfarben direkt nebeneinander sind (gerade, nicht schräg).
Dabei vergibt er Punkte:

- Rot neben Gelb gibt 3 Punkte.
- Gelb neben Orange gibt 1 Punkt.
- Ansonsten gibt es keine Punkte.

Am Ende zählt Peter alle Punkte zusammen.
Die Bepflanzung ist perfekt, wenn sie so viele Punkte bekommt wie möglich.
Außerdem muss jede Blumenfarbe mindestens einmal vorkommen.

Finde die perfekte Bepflanzung!



So ist es richtig:
Ben vergibt Punkte für benachbarte Blumen-Paare. Jede Linie im Blumenkasten-Bild steht für ein solches Paar. Eine Bepflanzung bekommt möglichst viele Punkte (und ist dann perfekt), wenn so viele Paare wie möglich aus einer roten und einer gelben Blume bestehen. Es muss aber auch mindestens eine orange Blume vorkommen. Um auch mit einer orangen Blumen Punkte zu machen, muss sie mit gelben Blumen Paare bilden. Es sollte aber nur so wenige orange-gelbe Paare geben wie möglich, damit so viele Linien bzw. Paare für die Kombination rot-gelb übrig bleiben. Das kann man erreichen, wenn man eine einzige orange Blume in eine Ecke platziert – egal in welche. Dann kann sie nur an zwei Paaren beteiligt sein; weniger geht nicht. Neben die orange Blume werden dann gelbe Blumen gesetzt und dann abwechselnd rote und gelbe, wie im Bild. Diese Bepflanzung ist perfekt, denn mehr Punkte kann eine Bepflanzung nicht bekommen.

Das ist Informatik!
Das Bestreben nach *Optimierung* hat Menschen schon immer ausgezeichnet: der Wunsch, für ein Problem nicht nur irgendeine Lösung zu finden, sondern die beste. Optimierung ist alltäglich: Wer einen bestimmten Computer kaufen will, sucht nach dem Angebot mit dem niedrigsten Preis. Wer von A nach B reisen will, möchte nicht irgendeine Route nehmen, sondern die günstigste. Aber was bedeutet „günstig" in diesem Fall: Soll die Route möglichst geringe Kosten verursachen, eine möglichst geringe Länge in km haben oder möglichst wenig Zeit in Anspruch nehmen?

Wenn man ein Optimierungsproblem lösen will, benötigt man also nicht nur die Kenntnis, wie eine Lösung des Problems überhaupt aussieht. Man braucht euch eine Funktion, die den Wert einer Lösung berechnen kann. Eine optimale Lösung (oder auch: ein Optimum) ist dann eine Lösung mit dem höchsten Wert – oder mit dem niedrigsten Wert, je nach Funktion. In dieser Biberaufgabe ist Bens Methode zur Berechnung eines Punktwerts für ein Blumenarrangement eine solche Optimierungsfunktion.

In der Mathematik wird Optimierung seit Jahrhunderten untersucht. In der Informatik wurde viel Sorgfalt darauf verwandt, mathematische Optimierungsmethoden als Algorithmen zu implementieren: Computer können für die Optimierung bei groß angelegten Problemen eingesetzt werden, bei denen Menschen keinen Erfolg hätten. Es ist bekannt, dass viele Optimierungsprobleme sehr schwer zu lösen sind. Informatikerinnen und Informatiker haben deshalb für solche Probleme Methoden (sogenannte Heuristiken) erfunden und untersucht, die nicht immer das Optimum, aber Lösungen nahe am Optimum finden.

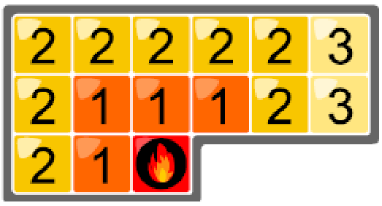
3-4: – 5-6: – 7-8: schwer 9-10: mittel 11-13: leicht

Hotspot-Heizung

Luis mag es warm im Bad. In sein neues Haus lässt er eine Bodenheizung mit Hotspots einbauen.

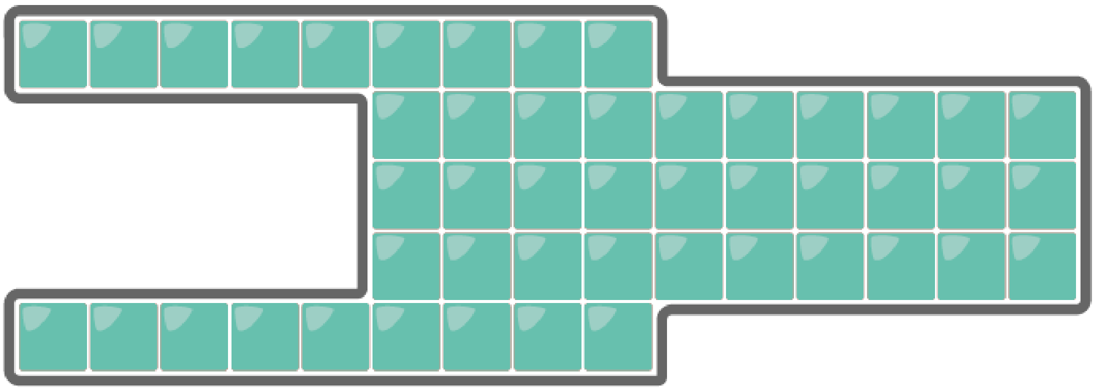
Ein Hotspot wird direkt unter einer Fliese montiert.
Schaltet man den Hotspot ein, wird diese Fliese sofort warm. Von einer warmen Fliese breitet sich die Wärme in einer Minute auf alle benachbarten Fliesen aus – seitlich und über Eck.

Hier ist ein Beispiel. Die Zahlen sagen für jede Fliese, nach wie vielen Minuten sie warm ist.

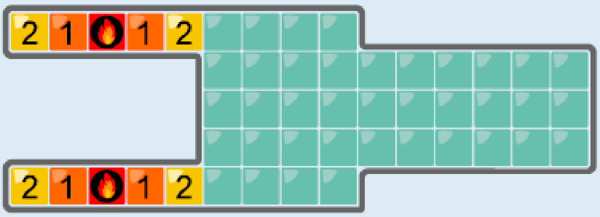


Luis kann sich 4 Hotspots leisten.
Luis wünscht, dass nach möglichst wenigen Minuten alle Fliesen im Bad warm sind.
Die Hotspots werden gleichzeitig angeschaltet.

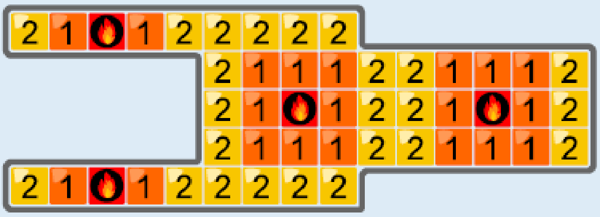
Hier ist das neue Bad.
Montiere die 4 Hotspots so, dass Luis' Wunsch erfüllt wird.



So ist es richtig:
Man kann die 4 Hotspots  so montieren, dass alle Fliesen des Badezimmers nach 2 Minuten warm sind. Jeder Hotspot kann in der ersten Minute höchstens 9 Fliesen erwärmen, nach 2 Minuten höchstens 25 Fliesen. Vier Hotspots erwärmen in der ersten Minute höchstens 36 Fliesen, in 2 Minuten höchstens 100 Fliesen. Das Badezimmer hat 48 Fliesen. Es ist also unmöglich, mit vier Hotspots alle Fliesen in nur einer Minute zu erwärmen, aber 2 Minuten könnten ausreichen.
Nun müssen die Hotspots so montiert werden, dass alle Fliesen nach 2 Minuten warm sind. Wegen der schlechten Aufteilung des Raumes bietet es sich an, zwei Hotspots so zu montieren, dass die Fliesen in den beiden Gängen nach 2 Minuten warm sind:



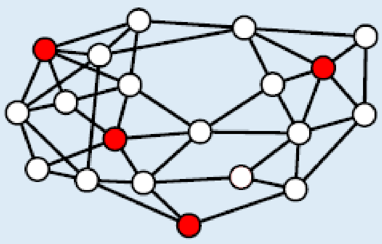
Die verbleibenden zwei Hotspots kann man dann so montieren:



Den ganz rechts montierten Hotspot kann man auch eine Fliese darüber oder darunter montieren. Auch dann sind nach 2 Minuten alle Fliesen warm.

Das ist Informatik!
Das möchte man häufig: mit möglichst wenig Aufwand möglichst viel erreichen. In dieser Biberaufgabe soll mit möglichst wenig Heizungen das Bad möglichst schnell warm werden. Will man ein großes Haus mit WLAN ausstatten, möchte man mit möglichst wenigen Hotspots eine gute Abdeckung aller Räume erzielen. Es gibt noch viele andere Beispiele.
In der Informatik ist ein ähnliches Problem bekannt, nämlich die Bestimmung eines minimalen „Dominating Set“ in einem Graphen. Ein Graph ist eine Struktur aus Knoten und Kanten (das sind Paare von Knoten). Die beiden Knoten einer Kante nennt man auch Nachbarn. Ein Dominating Set (DS) ist eine Teilmenge der Knoten, für die gilt: Jeder andere Knoten muss einen Nachbarn haben, der im Dominating Set enthalten ist. Ein DS ist minimal, wenn es keine kleinere Knoten-Teilmenge gibt, die auch ein DS ist.

Hier ist ein Beispiel-Graph: Die Knoten sind als Kreise, die Kanten als Linien gezeichnet. Jeder weiße Knoten ist Nachbar mindestens eines roten Knotens. Also sind die roten Knoten ein DS, und zwar ein minimaler: Es gibt kein DS mit weniger als vier Knoten.
Die Platzierung der Heizungen kann man als Bestimmen eines minimalen DS verstehen: Die Fliesen sind die Knoten. Zwei Fliesen sind „Heizungs-Nachbarn“ (bilden also eine Kante), wenn das Montieren einer Heizung unter einer Fliese bedeutet, dass die andere nach höchstens zwei Minuten warm ist. Fliesen, die in diesem Graphen ein minimales DS bilden, stellen eine Lösung dieser Biberaufgabe dar.



3-4: –	5-6: –	7-8: –	9-10: schwer	11-13: mittel

Durch den Tunnel

Anna und Benno machen mit ihren Eltern eine Wanderung. Auf ihrer Strecke liegt ein Tunnel. Aus Erfahrung wissen sie, dass jeder von ihnen unterschiedlich viel Zeit für die Tunnelpassage benötigt: Anna benötigt 10 Minuten, Benno 5 Minuten, die Mutter 20 Minuten und der Vater 25 Minuten.

Den dunklen und engen Tunnel kann man nur alleine oder zu zweit passieren. Sie müssen also mehrere Passagen machen. Zu zweit benötigt man so viel Zeit wie die langsamere der beiden Personen. Im Tunnel muss man auf jeden Fall eine Lampe benutzen.

Als sie an den Eingang des Tunnels kommen, stellen sie fest: Der Akku ihrer einzigen Lampe reicht nur noch für 60 Minuten. Können sie innerhalb dieser 60 Minuten alle durch den Tunnel kommen?

Anna hat einen Plan: „Ja, können wir, und zwar mit fünf Passagen!“

Ziehe die Namen so in die passenden Felder, das Annas Plan umgesetzt wird.

Hin →		
← **Zurück**		
Hin →		
← **Zurück**		
Hin →		

Musteraufgaben Stufe 9-10

So ist es richtig:
Dass alle innerhalb von 60 Minuten am Ausgang sind, ist nur möglich, wenn Mutter und Vater (die beiden langsamsten) nur einmal den Tunnel passieren. Das wiederum ist nur möglich, wenn vorher Anna und Benno den Tunnel passieren und einer von ihnen zurück geht. Der eine kann dann die Lampe Vater und Mutter geben, und der andere bringt sie durch den Tunnel zurück, damit dann beide rechtzeitig zusammen zum Ausgang gehen können. Dabei ist es egal, ob in Passage 2 Anna zurückgeht oder Benno: In Passage 4 ist der jeweils andere dran, so dass diese beiden Passagen in beiden Fällen zusammen 15 Minuten dauern.

Passage	Person 1	Person 2	Minuten
Hin	Anna 10 min	Benno 5 min	10
Zurück	Benno 5 min		15
Hin	Mutter 20 min	Vater 25 min	40
Zurück	Anna 10 min		50
Hin	Anna 10 min	Benno 5 min	60

Passage	Person 1	Person 2	Minuten
Hin	Anna 10 min	Benno 5 min	10
Zurück	Anna 10 min		20
Hin	Mutter 20 min	Vater 25 min	45
Zurück	Benno 5 min		50
Hin	Anna 10 min	Benno 5 min	60

Das ist Informatik!
Anna hat die Lösung des Tunnelproblems mit einem Plan beschrieben. Ihr Plan erfüllt die Bedingung, dass bei jeder Passage die Lampe dabei ist. Die Ausführung des Plans kommt mit den verfügbaren Mitteln aus: der 60-Minuten Akku-Reserve der Lampe.

Die Informatik entwickelt immer komplexere Systeme, die in der Lage sind, eigenes Handeln selbständig bzw. autonom zu planen, dabei die vorgegebenen Bedingungen zu erfüllen und mit den verfügbaren Mitteln auszukommen; so wie Anna. Die Konsequenzen autonomen Planens sind weitreichender, als in unserer gut überschaubaren Biberaufgabe. Zum Beispiel bei selbstfahrenden Autos: Die müssen einen sinnvollen Weg zum Ziel planen, dabei alle Verkehrsregeln einhalten und berücksichtigen, dass sie während der Ausführung immer genug Energie haben.

Souveräne und sichere Informatik-Planung wird auch ethisch und juristisch vertretbar sein müssen, wenn das Handeln autonomer Systeme Auswirkungen für Menschen, andere Lebewesen und die Umwelt hat. Ein selbstfahrendes Auto muss stets die zentrale Bedingung erfüllen, dass durch sein Handeln kein Mensch zu Schaden kommt – wenn irgend möglich. Solche Überlegungen hat Isaac Asimov bereits im Jahr 1942 angestellt, als gerade die ersten Computer gebaut wurden. Er hat sich Gesetze für Roboter überlegt. Sein erstes Robotergesetz lautet: „Ein Roboter darf keinen Menschen verletzen oder durch Untätigkeit zu Schaden kommen lassen." Und ein selbstfahrendes Auto ist ein Roboter.

Weitere Biber-Aufgaben findest du hier:
https://bwinf.de/biber/archiv/aufgabensammlung/