

# Die Aufgaben der 2. Runde

## Allgemeine Hinweise

Herzlichen Glückwunsch zum Erreichen der 2. Runde! Hier sind die Aufgaben. Sie sind anspruchsvoll, und ihre Bearbeitung ist aufwändig. Aber die Mühe lohnt sich, denn durch Teilnahme an der 2. Runde

- wirst du sicher sehr viel lernen;
- kannst du dich für die Endrunde qualifizieren;
- kannst du einen Buchpreis der Verlage O'Reilly oder dpunkt.verlag gewinnen;
- hast du am Ende eine Arbeit fertig gestellt, die du als Besondere Lernleistung in die Abiturwertung einbringen kannst;
- kannst du dich (als jüngerer Teilnehmer) um die Teilnahme an einer Deutschen Schülerakademie bewerben;
- hast du die Chance auf eine Einladung zu den „Forschungstagen Informatik 2019“ des Max-Planck-Instituts für Informatik in Saarbrücken.

Wir wünschen also viel Spaß und viel Erfolg bei der Bearbeitung!

Es gibt drei Aufgaben. **Eine Einsendung darf Bearbeitungen zu höchstens zwei Aufgaben enthalten**, deren Bewertung dann das Gesamtergebnis ausmacht. Sollte eine Einsendung Bearbeitungen zu allen drei Aufgaben enthalten, werden wir zwei davon zufällig auswählen und nur diese bewerten.

An dieser Runde dürfen nur Einzelpersonen teilnehmen, die in der 1. Runde in drei Aufgaben insgesamt mindestens 12 Punkte erreicht oder einem Team angehört haben, dem dieses gelungen ist. Gruppenarbeit ist in der 2. Runde nicht zulässig.

**Einsendeschluss ist Montag, der 29. April 2019.**

## Bearbeitung

Die Bearbeitung einer Aufgabe sollte zunächst eine nachvollziehbare und vollständige Lösung aller Teilaufgaben enthalten. **Zusatzpunkte** für eine höhere Bewertung kannst du erreichen, wenn du die Aufgabe dort, wo es möglich und sinnvoll ist, eigenständig weiterentwickelst. Sinnvoll sind inhaltliche Erweiterungen und Verbesserungen, etwa von Datenstrukturen und Algorithmen; uninteressant sind aufwändige Tricks, z. B. zur reinen Verschönerung der Benutzeroberfläche. Begründe für jede Erweiterung, weshalb sie sinnvoll ist und ihre Realisierung eine eigene Schwierigkeit darstellt.

Grundsätzlich gelten die Vorgaben der 1. Runde weiter. Wesentliches Ergebnis der Aufgabebearbeitung ist also eine **Dokumentation**, in der du den *Lösungsweg* sowie die *Umsetzung* des Lösungswegs in das dazugehörige Programm beschreibst. Die Beschreibung des Lösungswegs

kann mit Hilfe (halb-)formaler Notationen präzisiert werden, die Beschreibung der Umsetzung mit Verweisen auf die entsprechenden Quellcode-Elemente.

In die Dokumentation gehören auch *Beispiele* (Programmein- und -ausgaben oder Zwischenschritte), die zeigen, wie das Programm sich in unterschiedlichen Situationen verhält. Komplettiert wird die Dokumentation durch *Auszüge aus dem Quelltext*, die alle wichtigen Module, Methoden, Funktionen usw. enthalten.

Weiteres Ergebnis der Bearbeitung ist die **Implementierung**. Sie besteht aus dem zur Lösung der Aufgabe geschriebenen lauffähigen *Programm* und dem vollständigen *Quelltext*. Außerdem können Beispielein- und -ausgaben oder weiteres hilfreiches Material der Implementierung beigefügt werden.

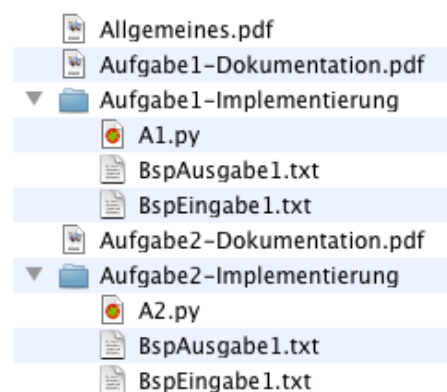
Die Dokumentation zu einer Aufgabe mit allen oben genannten Bestandteilen muss als PDF-Dokument eingereicht werden. Dieses Dokument wird für die Bewertung ausgedruckt. **Es kann sein, dass für die Bewertung deiner Einsendung nur die Dokumentation herangezogen wird.** Sie sollte also einen lückenlosen und verständlichen Nachweis des Leistungsumfangs und der Funktionstüchtigkeit der Programme geben – und unbedingt auch Beispiele enthalten!

Der Umfang der Dokumentation soll sich in Grenzen halten; eine gute Dokumentation vermittelt kurz und präzise alles Nötige, insbesondere die wesentlichen Ideen beim Lösungsweg. Nötig ist alles, was Interessierte mit guten Informatikkenntnissen, die die Aufgabenstellung kennen, wissen müssen, um den Lösungsweg zu verstehen und seine Umsetzung nachzuvollziehen.

Entscheidend für eine gute Bewertung sind zwar richtige (und sauber umgesetzte) Lösungswege, aber die Dokumentation hat schon oft den Ausschlag für oder gegen das Weiterkommen gegeben. Daher kann es nützlich sein, wenn das Erstellen der Dokumentation die Programmierarbeit eng begleitet. Außerdem ist unsere Erfahrung, dass ein Teilnehmer, der seine Lösungsidee nicht verständlich formuliert, meist auch keine fehlerlose Implementierung schafft. So kann es hilfreich sein, die Verständlichkeit der Dokumentation von Dritten prüfen zu lassen, selbst wenn sie fachfremd sind.

## Einsendung

Die Einsendung erfolgt wieder über das BWINF-PMS ([pms.bwinf.de](http://pms.bwinf.de)). Hochladen kannst du ein max. 40 MB großes ZIP-Archiv (z. B. `VornameNachname.zip`); sein Inhalt sollte so strukturiert sein wie rechts abgebildet. Die Dokumentationen der bearbeiteten Aufgaben müssen als PDF-Dokumente enthalten sein; Dateien in anderen Formaten werden möglicherweise ignoriert. Ein Dokument `Allgemeines.pdf` ist nur dann nötig, wenn du allgemeine, von den Aufgabebearbeitungen unabhängige Bemerkungen zu deiner Einsendung machen willst. Die Schriftgröße einer Dokumentation muss mindestens 10 Punkt sein, bei Quelltext mindestens 8 Punkt. Auf jeder Seite einer Dokumentation sollen in der Kopfzeile die Teilnahme-Id, Vorname, Name und Seitennummer stehen; hierfür sind auf der BWINF-Webseite entsprechende Dateien als Vorlage. Die Teilnahme-Id steht auf der Teilnahmebescheinigung der 1. Runde.



## Weitere Hinweise

Bei der Bewertung können Programme unter Windows (7 / 10), Linux, Mac OS X (10.12) und Android ausgeführt werden.

**Fragen zu den Aufgaben** können per Mail an [bundeswettbewerb@bwinf.de](mailto:bundeswettbewerb@bwinf.de) oder telefonisch unter 0228-378646 (zu üblichen Arbeitszeiten) gestellt werden. Die Antwort auf E-Mail-Anfragen kann sich leicht verzögern. Informationen zur 2. Runde finden sich auf unseren Webseiten ([bwinf.de/bundeswettbewerb](http://bwinf.de/bundeswettbewerb)). In der Community von [einstieg-informatik.de](http://einstieg-informatik.de) werden sicher wieder viele Teilnehmer über die Aufgaben diskutieren – ohne Lösungsideen auszutauschen.

Allen Teilnehmern der 2. Runde wird bis Ende Juni 2019 die Bewertung mitgeteilt. Die Besten werden zur Endrunde eingeladen, die 8.–11. Oktober 2019 von SAP Deutschland in Walldorf ausgerichtet wird. Dort werden die Bundessieger und Preisträger ermittelt und ausgezeichnet. Bundessieger werden in der Regel ohne weiteres Auswahlverfahren in die Förderung der Studienstiftung des deutschen Volkes aufgenommen. Außerdem werden Geld- und Sachpreise vergeben. Der Rechtsweg ist ausgeschlossen.

Viel Spaß und viel Erfolg!

## Aufgabe 1: Lisa rennt

Lisa schläft gern, besonders morgens. So schafft sie es nicht immer, rechtzeitig von daheim loszugehen, um an der Haltestelle zu sein, wenn der Bus um 7.30 Uhr von dort abfährt.

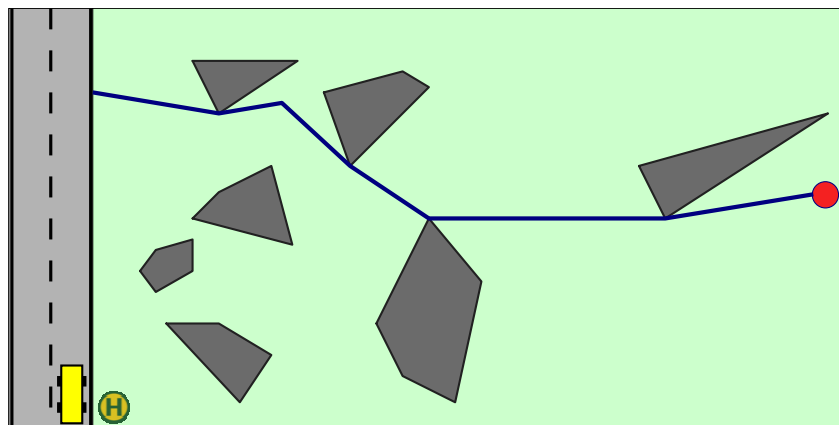
Zum Glück kennt sie der Busfahrer, und er nimmt sie auch mit, falls sie irgendwo am Rand der schnurgeraden Landstraße steht, wenn der Bus dort vorbeikommt. Da der Bus nur 30 km/h fährt und sie selbst flotte 15 km/h über eine beträchtliche Strecke schafft, kann sie durch geschicktes Querfeldeinlaufen etwas Zeit herausschlagen.

Dabei muss sie allerdings Hindernisse wie Baustellen und Tümpel umlaufen. Heute will sie es endlich mal genau wissen: Was ist der letztmögliche Zeitpunkt, an dem sie ihr Haus spätestens verlassen muss, um den Bus irgendwo gerade noch zu erreichen? Welche Route soll sie dabei einschlagen?

Lisa wünscht sich ein Programm, das ihr diese Frage beantwortet. Hierfür überlegt sie sich zuerst, wie sie laufen würde, wenn es keine Hindernisse gäbe.

### Aufgabe

Schreibe ein Programm, das das Problem allgemein löst. Als Eingabe erhält das Programm die Koordinaten von Lisas Haus sowie der als Polygone modellierten Hindernisse. Dabei wird ein Koordinatensystem zugrunde gelegt, in dem der Bus die  $y$ -Achse abfährt, die Haltestelle am Punkt  $(0,0)$  ist und alle Längen in Metern angegeben werden.



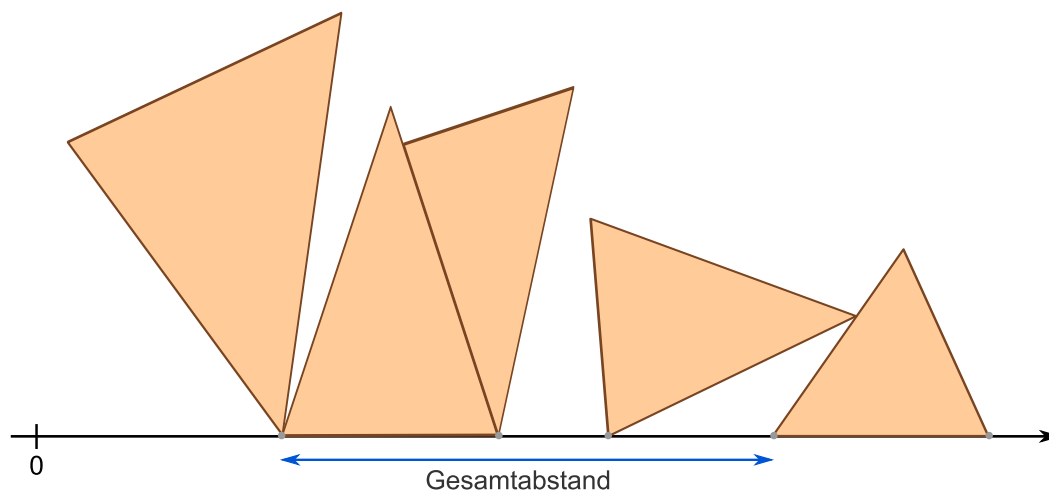
In der Abbildung siehst du für ein Beispiel eine mögliche Route an verschiedenen Hindernissen vorbei von Lisas Haus zur Landstraße, die aber offensichtlich nicht besonders gut ist.

Wende dein Programm auf alle Beispiele an, die du auf den BWINF-Webseiten findest; das Datenformat ist dort erläutert. Dokumentiere die Ergebnisse.

## Aufgabe 2: Dreiecksbeziehungen

Der seltsame Klub der wohlhabenden Trianguläre will entlang der schnurgeraden Küstenstraße auf der Landseite eine neue Siedlung gründen. Jedes Klubmitglied hat sich für eine dreieckige Grundstücksform entschieden, deren Größe und Winkel seine Persönlichkeit zum Ausdruck bringen. Jetzt gilt es, die Grundstücke entlang der Straße so unterzubringen, dass jedes Grundstück mindestens eine Ecke an der Straße hat, damit der Zugang zum Grundstück gesichert ist.

Alle Ecken eines Grundstücks an der Straße heißen Straßenecken. Der Abstand zwischen zwei Grundstücken ist die kleinste Distanz zwischen einer Straßenecke des einen und einer Straßenecke des anderen Grundstücks. Die Klubmitglieder möchten so eng beieinander wohnen, wie es ihre Grundstücke erlauben. Daher soll der Abstand zwischen den am weitesten voneinander entfernten Grundstücken (Gesamtabstand) möglichst gering sein. Wie die Trianguläre entdecken, ist es gar nicht so leicht, die optimale Platzierung aller Grundstücke herauszufinden. Im hier abgebildeten Beispiel lässt sich der Gesamtabstand bestimmt noch weiter verringern.



### Aufgabe

Schreibe ein Programm, das eine Liste von Grundstücken in Dreiecksform einliest und sie so platziert, dass eine Ecke oder eine Kante jedes Dreiecks an der Küstenstraße (der  $x$ -Achse des Koordinatensystems) liegt und der Gesamtabstand möglichst klein ist.

Das Programm soll Folgendes ausgeben: (a) eine Visualisierung der Lage der Dreiecke entlang der  $x$ -Achse wie im hier abgebildeten Beispiel sowie (b) die berechnete Folge der Dreiecke, aufsteigend sortiert nach ihrer Platzierung entlang der  $x$ -Achse, mit den Koordinatenpaaren  $(x,y)$  aller drei Ecken jedes Grundstücks.

Wende dein Programm auf alle Beispiele an, die du auf den BWINF-Webseiten findest; das Datenformat ist dort erläutert. Dokumentiere die Ergebnisse.

## Aufgabe 3: Schach dem Wildschwein

Diese Aufgabe hat zwei gleichwertige Varianten A und B. Wenn du die Aufgabe bearbeitest, entscheide dich für eine von ihnen. Die zweite Variante richtet sich eher an Schachfreunde.

### Variante A

Julian hat auf einem Flohmarkt ein interessantes Brettspiel für zwei Personen gefunden. Das Spielbrett hat  $8 \times 8$  Felder. Die Spielfiguren sind ein Wildschwein, drei Jagdhunde und ein Jäger. Ein Spieler hat das Wildschwein als Spielfigur und der andere den Jäger und die Jagdhunde. Für das Spiel gelten folgende Regeln:

1. Die Spieler müssen abwechselnd mit genau einer eigenen Spielfigur ziehen.
2. Der Jäger wie auch das Wildschwein darf sich in einem Spielzug auf ein angrenzendes Feld bewegen (horizontal, vertikal oder diagonal).
3. Ein Jagdhund darf in einem Spielzug auf ein Feld springen, das zwei Felder horizontal und ein Feld vertikal oder zwei Felder vertikal und ein Feld horizontal entfernt ist.
4. Es ist verboten, eine Spielfigur auf ein Feld zu setzen, auf dem bereits eine eigene Spielfigur steht.
5. Wenn ein Spieler seine Figur auf ein Feld setzt, auf dem bereits eine gegnerische Figur steht, hat er die gegnerische Figur gefangen und das Spiel gewonnen.
6. Das Wildschwein darf sich auf kein Feld bewegen, auf dem es im nächsten Zug gefangen werden kann, es sei denn, es geht nicht anders.

Nachdem Julian sich lange mit dem Spiel beschäftigt hat, fragt er sich, ob der Spieler mit dem Jäger immer gewinnen kann, egal wie die Figuren stehen. Er stellt schnell fest, dass dies nicht immer geht, zum Beispiel wenn die drei Jagdhunde direkt neben dem Wildschwein stehen. Als Nächstes fragt er sich, ob der Spieler mit dem Jäger auf eine bestimmte Weise gewinnen kann.

### Aufgabe (Variante A)

Schreibe ein Programm, das für eine gegebene Stellung der Figuren und ein gegebenes Zielfeld feststellt, ob der Spieler, der den Jäger hat, **das Wildschwein auf dieses Zielfeld zwingen kann, so dass er es dort mit der Gefangennahme bedroht und schließlich durch seinen nächsten Zug fängt und dadurch gewinnt.**

Wende dein Programm auf alle Beispiele an, die du auf den BWINF-Webseiten findest; das Datenformat ist dort erläutert. Dokumentiere die Ergebnisse und Spielzüge, so dass sie nachvollziehbar sind; nur Ja/Nein-Antworten genügen nicht.

**Zudem ist folgendes Problem zu lösen: Das Wildschwein steht in der obersten Reihe des Spielbretts und die anderen Figuren stehen in der untersten Reihe. Für welche Zielfelder kann der Spieler, der den Jäger hat, es erzwingen, dass er das Wildschwein durch seinen nächsten Zug fängt und gewinnt? Es ist nur bekannt, dass dies für die Eckfelder des Spielbretts möglich ist.**

## Variante B

Schach ist ein komplexes Spiel, bei dem es eine sehr große Anzahl an möglichen Spielverläufen gibt. Obwohl sich seit Jahrhunderten schlaue Köpfe mit dem Spiel beschäftigen, ist es noch nicht vollständig analysiert. Es ist auch deshalb so komplex, weil es sechs verschiedene Arten von Spielfiguren gibt, die sich unterschiedlich auf dem Spielbrett bewegen können.

In dieser Aufgabe wollen wir einer Frage nachgehen, die sich nur mit zwei verschiedenen Typen von Spielfiguren beschäftigt: König und Springer. Schach-Vorwissen ist dazu nicht vonnöten, da wir alle hierfür benötigten Regeln kurz erklären:

Schach ist ein Spiel für zwei Personen, die abwechselnd ziehen. Eine Person spielt mit Weiß, die andere mit Schwarz. Das Spielbrett hat  $8 \times 8$  Felder.

In jedem Zug muss ein Spieler mit genau einer Figur seiner Farbe ziehen. Zieht er mit dem König, so muss dieser auf eins der angrenzenden Felder gesetzt werden (vertikal, horizontal oder diagonal). Zieht er mit einem Springer, so muss dieser auf ein Feld gesetzt werden, das entweder zwei Felder horizontal und ein Feld vertikal entfernt ist oder zwei Felder vertikal und ein Feld horizontal. Eine Spielfigur darf nicht auf ein Feld gesetzt werden, das von einer Figur derselben Farbe besetzt ist.

Ein Spieler kann eine Spielfigur des Gegners *schlagen*, d. h. aus dem Spiel nehmen, wenn er in seinem Zug eine eigene Figur auf das Feld setzt, auf dem die gegnerische Figur steht. Der König darf nicht auf ein Feld gesetzt werden, auf dem er im nächsten Zug vom Gegner geschlagen werden könnte. Setzt ein Spieler eine Figur auf ein Feld, von dem aus sie den König des Gegners schlagen könnte, so steht dieser König im *Schach* und muss im nächsten Zug versetzt werden. Ist dies nicht möglich, so ist der Gegner *matt* und hat verloren. Ist ein Spieler am Zug und hat keinen erlaubten Zug, so endet die Partie ebenfalls und das Spielergebnis ist *patt*.

Schachenthusiasten beschäftigen sich gern mit besonderen Spielsituationen und versuchen, sie vollständig zu verstehen. In dieser Aufgabe darfst du dich mit einem bestimmten Schachproblem beschäftigen: Nur der schwarze König, der weiße König und drei weiße Springer befinden sich auf dem Brett. Schwarz steht nicht im Schach und Weiß ist am Zug. Kann Weiß den schwarzen König matt setzen? Es ist bekannt, dass dies unter bestimmten Bedingungen möglich ist.

### Aufgabe (Variante B)

Schreibe ein Programm, das für eine gegebene Stellung und ein Zielfeld feststellt, ob Weiß erzwingen kann, dass der schwarze König auf diesem Zielfeld matt gesetzt wird.

Wende dein Programm auf alle Beispiele an, die du auf den BWINF-Webseiten findest; das Datenformat ist dort erläutert. Dokumentiere die Ergebnisse und Spielzüge, so dass sie nachvollziehbar sind; nur Ja/Nein-Antworten genügen nicht.

Zudem ist folgendes Schachproblem zu lösen: Der schwarze König steht in der obersten Reihe des Spielbretts und die weißen Figuren in der untersten Reihe. Für welche Felder kann Weiß erzwingen, dass der schwarze König dort matt gesetzt wird? Dieses Schachproblem ist noch offen! Es ist nur bekannt, dass es für die Eckfelder des Spielbretts möglich ist.

