

# Portfolioprüfung – Werkstück A – Alternative 5

## 1 Aufgabe

Entwickeln und implementieren Sie ein Bash-Skript, mit dem Benutzer in der Shell Sudoku spielen können.

Wikipedia enthält zahlreiche Informationen zu Sudoku<sup>1</sup> und den mathematischen Grundlagen<sup>2</sup>.

## 2 Anforderungen

- Ein 9x9 Felder großes, (lösbares) Sudoku wird mit einigen vorgegeben Zahlen erzeugt und in der Shell angezeigt bzw. ausgegeben.
- Wenn Sie es nicht schaffen, einen Algorithmus zur finden bzw. für die Shell anzupassen, der lösbarer Sudokus generiert, dann können Sie alternativ eine Lösung implementieren, bei der Sudokus aus einer oder mehreren Textdateien eingelesen werden (können). Das feste einprogrammieren von Sudokus in den Quellcode ist keine zulässige Lösung.
- **Entwickeln und implementieren Sie Ihre Lösung als Bash-Skript** als freie Software (Open Source) und verwenden Sie hierfür ein Code-Repository, z.B. bei GitHub.
- Für die „grafische Darstellung“ und Bedienung in der Shell können Sie eine Bibliothek wie **dialog**<sup>3 4 5</sup> oder **Whiptail**<sup>6 7 8</sup> verwenden.
- Es soll möglich sein, das Spiel nur aus einer Shell heraus zu spielen. Es soll komplett in der Shell ablaufen!
- Die Benutzer können in die freien Felder Zahlen eingeben.
- Die Benutzer können selbst eingegebene Zahlen auch wieder löschen.
- Die Anwendung kontrolliert nach jeder Eingabe einer Zahl, ob das Sudoku-Feld konsistent ist. Ist das Feld nach der Eingabe nicht mehr konsistent, reagiert die Anwendung angemessen.

---

<sup>1</sup><https://de.wikipedia.org/wiki/Sudoku>

<sup>2</sup>[https://en.wikipedia.org/wiki/Mathematics\\_of\\_Sudoku](https://en.wikipedia.org/wiki/Mathematics_of_Sudoku)

<sup>3</sup>[http://openbook.rheinwerk-verlag.de/shell\\_programmierung/shell\\_007\\_007.htm](http://openbook.rheinwerk-verlag.de/shell_programmierung/shell_007_007.htm)

<sup>4</sup><https://www.linux-community.de/ausgaben/linuxuser/2014/03/mehr-komfort/>

<sup>5</sup><https://linuxkurs.spline.de/Ressourcen/Folien/Linux-Kurs-7.pdf>

<sup>6</sup>[https://en.wikibooks.org/wiki/Bash\\_Shell\\_Scripting/Whiptail](https://en.wikibooks.org/wiki/Bash_Shell_Scripting/Whiptail)

<sup>7</sup><https://saveriomiroddi.github.io/Shell-scripting-adventures-part-3/>

<sup>8</sup><https://www.dev-insider.de/dialogboxen-mit-whiptail-erstellen-a-860990/>

- Die Anwendung kontrolliert nach jedem Löschen einer Zahl, ob das Sudoku-Feld konsistent ist.
- Es soll zu jeder Zeit klar ersichtlich sein, welche Zahlen vorgegeben waren und welche Zahlen durch den Benutzer eingegeben wurden.
- Hat der Benutzer das Sudoku erfolgreich gelöst, soll dieses entsprechend angezeigt werden. Das kann beispielsweise durch eine Laufschrift geschehen, durch ein Blinken oder durch ein Invertieren der Farben in der Shell, etc.
- Sobald ein Sudoku geladen wurde, läuft eine Uhr, die anzeigt wie viel Zeit bislang vergangen ist. Hat der Benutzer das Sudoku gelöst, wird auch die vergangene Zeit ausgegeben.
- Der Quellcode soll durch Kommentare verständlich sein.
- Bearbeiten Sie die Aufgabe in Teams zu **3 Personen**.
- Schreiben Sie eine aussagekräftige und ansehnliche Dokumentation (Umfang: **8-10 Seiten**) über Ihre Lösung.
- Die Funktionalität der Lösung müssen Sie in der Übung demonstrieren. Bereiten Sie einen Vortrag mit Präsentationsfolien und eine Live-Demonstration (Umfang: **15-20 Minuten**) vor.