

Sudoku-Zahlenrätsel mit dem Computer lösen

Teilnehmer:

André Stenzel	Heinrich-Hertz-Oberschule
Birtan Özel	Herder-Oberschule
Marco Haderk	Herder-Oberschule
Janine Gastaldello	Andreas-Oberschule
Sebastian Köhler	Georg-Forster-Oberschule
Lisa Seidel	Georg-Forster-Oberschule

Gruppenleiter:

Stefan Körkel	Humboldt-Universität zu Berlin, Mitglied im DFG-Forschungszentrum MATHEON „Mathematik für Schlüsseltechnologien“
---------------	--

Wir haben ein Computerprogramm erstellt, mit dem man ein Sudoku-Problem eingeben, das Rätsel lösen oder die Unlösbarkeit feststellen und die Lösung ausgeben kann. Im Projekt wurden verschiedene Lösungsverfahren ausprobiert. Außerdem haben die Schüler verstanden, was dabei mathematisch passiert. Die Projektgruppe hat gemeinsam eine Software erstellt, dabei wurde die Programmentwicklung in einzelne Teilaufgaben (z.B. Datenstrukturen, Lösungsalgorithmen, graphische Ein- und Ausgabe) unterteilt.

Inhaltsverzeichnis

1	Was ist Sudoku?	3
2	Das Programm	3
2.1	Datenstruktur	3
3	Einzelne Funktionen des Programms	3
3.1	Überprüfung der Lösungen	3
3.2	Einfaches Lösen mit Kandidatenlisten	3
3.3	Lösen durch Eliminierung	4
3.4	Lösen durch systematisches Ausprobieren	4
4	Weitere Lösungsstrategien	4
4.1	Paartechniken	4
4.2	Tripeltechniken	5
4.3	X-Wing-Technik	6
4.4	Widerspruchstechniken	7
4.5	Die Nishio-Technik	8
5	Screenshots	9
6	Lösen eines Beispielsudokus	11

1 Was ist Sudoku?

- Sudoku ist ein Logikrätsel. Es besteht aus einem Gitterfeld mit 3x3 Blöcken, die jeweils in 3x3 Felder unterteilt sind, insgesamt also aus 81 Feldern in 9 Reihen und 9 Spalten. In einigen Feldern sind von vornherein Ziffern eingetragen.
- Ziel dieses Spiels ist es, alle Felder auszufüllen. Dabei dürfen in jeder Zeile, in jeder Spalte, in jedem Block nur einmal die Ziffern von 1-9 stehen.

2 Das Programm

2.1 Datenstruktur

Das Sudokuprogramm wurde objektorientiert programmiert. Es besteht aus einer Klasse(CSudokuBase).

CSudokuBase	Parent
array ([n][n]) kl[n][n][n]	
pruefen();widerspruch(); laden(); klerstellen(); sudokunachkloesen()	

Von der Klasse CSudokubase wurde die Klasse CSudoku abgeleitet, sie stellt zusätzlich eine grafische Oberfläche zur Verfügung.

CSudoku	Child
Widgeterstellen();Tabelleerstellen(int);speichern();laden()	

3 Einzelne Funktionen des Programms

3.1 Überprüfung der Lösungen

Durch Scannen jeder einzelnen Zeile, Spalte und jedes einzelnen Blockes wird in einer Tabelle die Häufigkeit des Auftretens der Zahlen von 1 bis 9 notiert und nur, wenn am Ende jede Zahl genau einmal auftritt, ist das Sudoku korrekt gelöst.

3.2 Einfaches Lösen mit Kandidatenlisten

Für jedes Feld werden die noch möglichen Zahlen aufgelistet. Dabei werden im Idealfall soviele Zahlen ausgeschlossen, dass man eine Zahl endgültig eintragen kann.

3.3 Lösen durch Eliminierung

Durch Scannen jeder einzelnen Zeile, Spalte und jedes einzelnen Blockes wird in einer Tabelle die Häufigkeit des Auftretens der Zahlen von 1 bis 9 notiert. Wenn eine Zahl nur noch einmal vorkommen kann, wird sie notiert.

3.4 Lösen durch systematisches Ausprobieren

- Das Feld mit der kleinsten Kandidatenliste wird rausgefiltert.
- Dort wird zuerst die kleinste der möglichen Kandidatenzahlen ausprobiert.
- Das Ziel ist, dadurch weitere Zahlen zu finden.
- Führt es auf einen Widerspruch, wird die nächste Zahl ausprobiert, bei der kein Widerspruch auftritt.

4 Weitere Lösungsstrategien

4.1 Paartechniken

4 7	4 7	1	4 7	3	4 8	6	2 5	2 5
			8 9				7	9

In der Abbildung sieht man eine Zeile aus einem Sudoku. In den beiden Feldern ganz links können nur noch die 4 und die 7 stehen. Das bedeutet, dass in einem der Felder die 4 und in einem die 7 steht, was zur Folge hat, dass in dieser Zeile sonst weder 4 noch 7 vorkommen.

4.2 Tripeltechniken

3 8	4 6 A	9	1 3 5	1 2 4	1 2 5	7	1 8	2 5 B 6
1	7			2 4 9	8	3		2 5 6
	2	5		7	6	4		9
7	1		8	6 9			2	3 4
5	4 8 B	2 8	7	1 2 A 6	1 2 3	1 8	9	3 4
9	3	2 8	4	1 2 B 6	1 2 5	1 8	7	5 8 A
2	5 9	7		3			4	8
4	5 9	3		8	7			1
6	8	1		5	4		3	7

In der Abbildung sieht man ein Sudoku. Betrachtet man nun nur die Zeilen 1, 5 und 6, dann sieht man, dass in jeder dieser Zeilen genau zweimal die 6 vorkommt. Sie kann entweder an die Stellen, die mit A bzw. B gekennzeichnet sind, eingesetzt werden, die anderen Stellen können somit aus der Auswahl entfernt bzw. die in der Kandidatenliste vermerkte 6 gestrichen werden.

4.3 X-Wing-Technik

6		5	2	C 7 9	8	4	1	D 3 7
7	2	4	3	1			8	
1		8	4	5				
8	5	6	1	3	4	7	2	9
3	4		9		5	1	6	8
9	1		8	6				4
	7	3	6	8			4	1
	6	9	7	4	1	8		
4	8	1	5	A 2 9	3	6		B 2 7

In der Abbildung sieht man ein Sudoku. In Feld A, dem Startfeld, kann noch eine 2 oder eine 9 stehen. Was wäre, wenn in diesem Feld eine 2 stünde? Dann stünde in Feld B eine 7 und in Feld D, dem Zielfeld eine 3. Fragt man hingegen was wäre, wenn im Startfeld eine 9 stünde. Dann stünde in Feld C eine 7 und in Feld D eine 3. Da im Startfeld keine andere Zahl als 2 und 9 stehen kann, kommt man zu dem Ergebnis: egal, ob in Feld A eine 2 oder 9 steht, im Zielfeld muss immer eine 3 stehen. Somit kann man da nun die 3 notieren.

4.4 Widerspruchstechniken

	9		8	7	5	6	4	2
2	5	4	9	6	1		F 7 8	3
8	6	7			4	5	1	9
5	C 3 7	9	D 2 3	4	8	1	E 2 7	6
4	1	2	7	5	6	9	3	8
	8	6	1		9		5	
	8	6	1		9		5	
	4	5	6		2		9	
6	A 2 3	5	9	7			G 2 8	

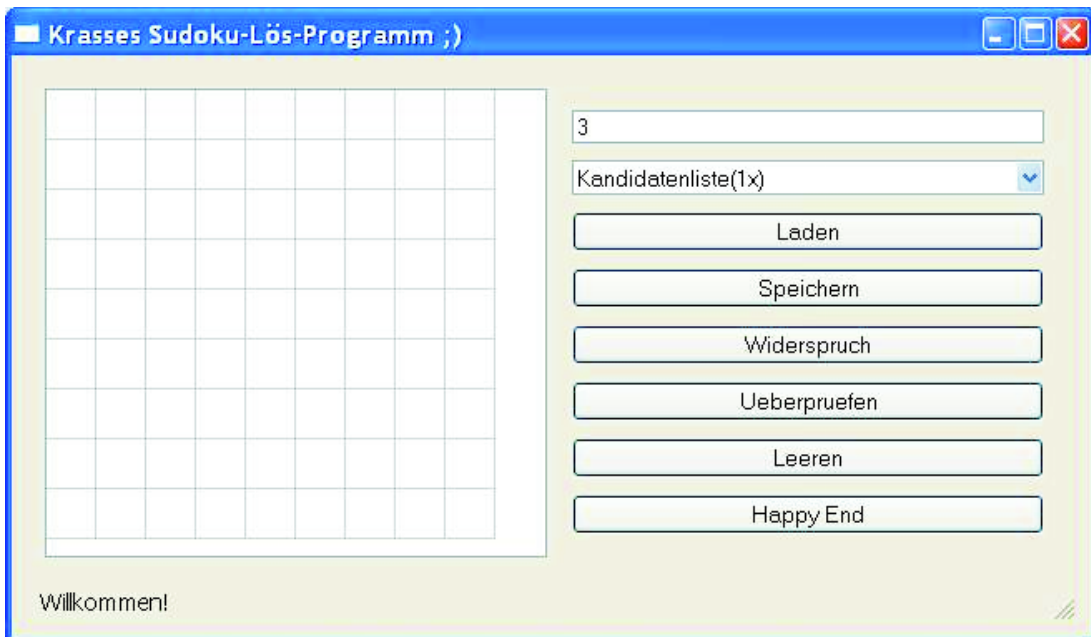
In dem Sudoku sind die relevanten Kandidatenlisten eingezeichnet, sie wurden mit den Buchstaben A bis G markiert. Die Technik funktioniert wie folgt: was wäre, wenn in Feld A, dem Startfeld, eine 2 steht? Dann stünde in Feld B eine 7 und im Feld C eine 3, daraus folgt dann weiter, dass in Feld D eine 2 stünde und in Feld E eine 7 und weiter folgt, dass in Feld F 8 stünde und Feld G eine 2 stünde. Damit befinden sich aber zwei Zweien in der Zeile. Das ist ein Widerspruch, deshalb kann im Startfeld keine 2 stehen.

4.5 Die Nishio-Technik

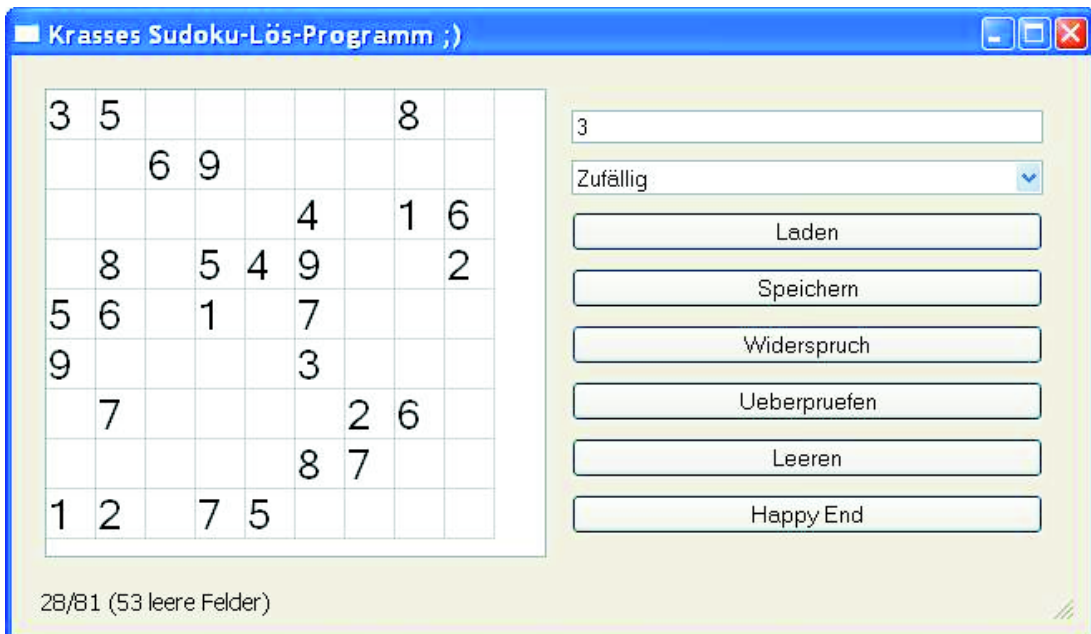
	9		8	7	5	6	4	2
2	5	4	9	6	1			3
8	6	7	2		4	5	1	9
5		9		4	8	1	2	6
4	1	2	7	5	6	9	3	8
	8	6	1	2	9		5	
	4	5	6		2		9	
6	2		5	9	7			
9			4		3	2	6	5

Ein Sonderfall dieser Methode. Als Beispiel stellt man sich wieder die Frage: was wäre, wenn im Feld A eine 2 stünde. Diesmal verfolgt man nur die Zweien in den anderen Gebieten. Man sieht dort: im Gebiet unten rechts gibt es nur einen Platz für die 2 und damit auch im Gebiet in der Mitte rechts und so weiter für alle anderen Gebiete.

5 Screenshots



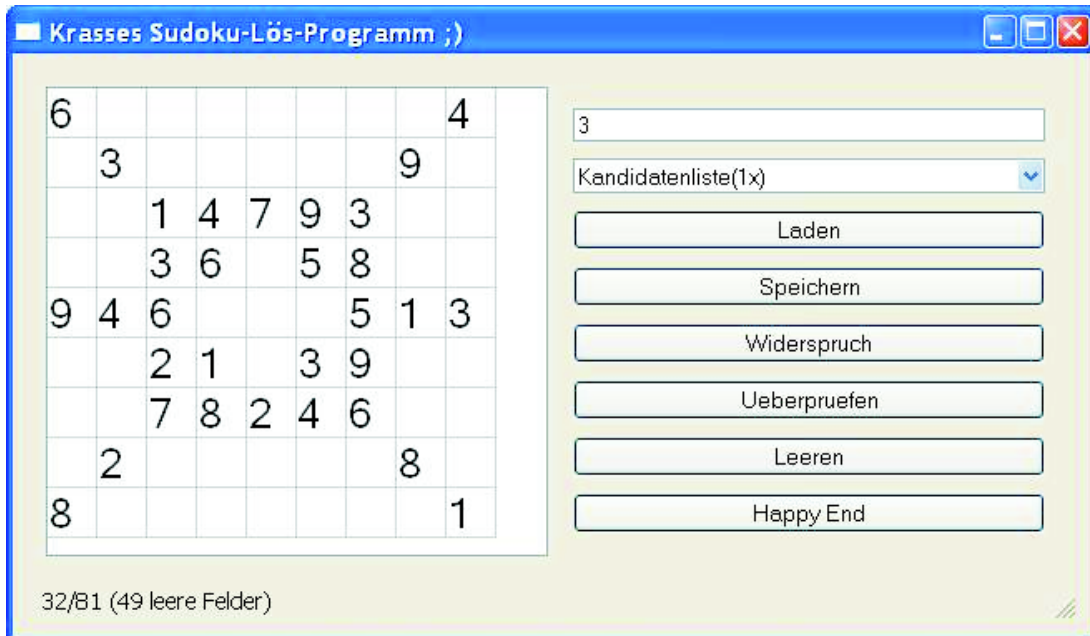
Auf dem Screenshot sieht man die Programmoberfläche, die das Sudokugitter und die verschiedenen Lösungsmöglichkeiten enthält. Weiterhin sind die Buttons mit ihren verschiedenen Aufgaben zu erkennen.



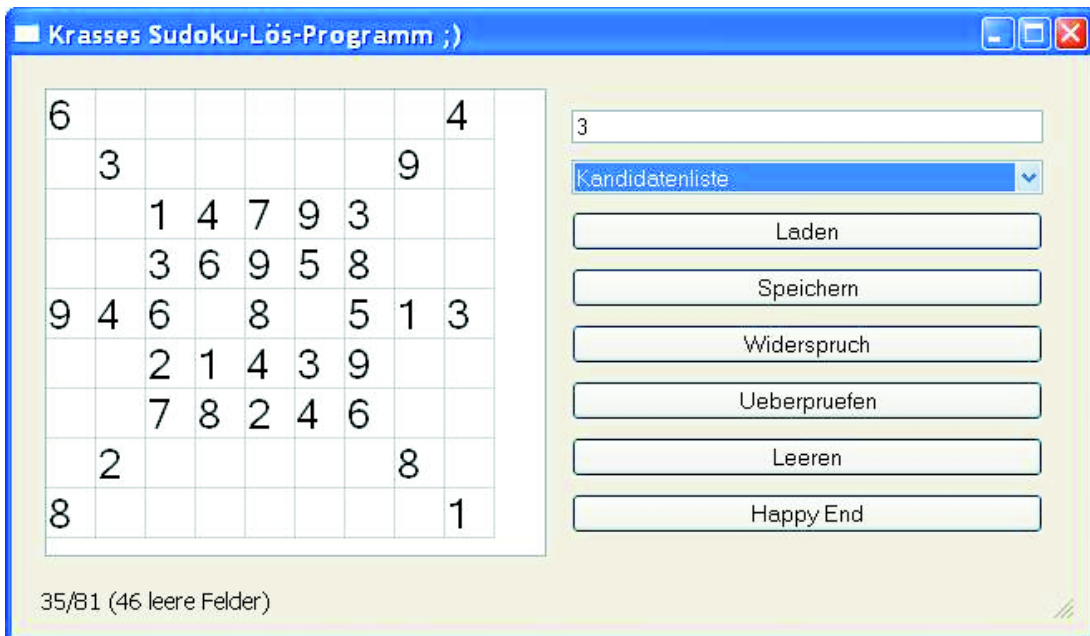
- Mit Hilfe des Ladenbuttons ist es möglich, Sudokus zu laden.
- Mit Hilfe des Speichernbuttons ist es möglich, gelöste Sudokus zusichern.

- Der Widerspruchbutton ermöglicht das Überprüfen des Sudokus, bevor man es löst.
- Nachdem man das Sudoku gelöst hat, kann man mit Hilfe des Überprüfenbuttons überprüfen, ob man es korrekt gelöst hat.
- Nach erfolgreichen Lösen kann man mit Hilfe des Leerenbuttons das Sudoku gitter schnell leeren.
- Mit dem Happy End-Button verlässt man das Programm.

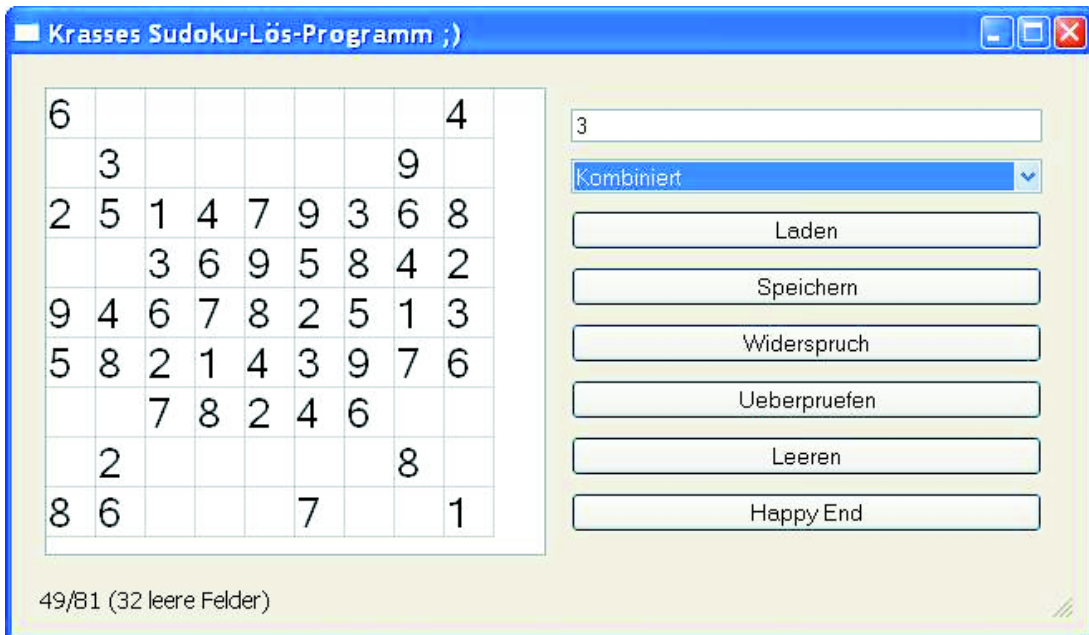
6 Lösen eines Beispielsudokus



Hier sieht man das Kamikazerätsel aus der Zeitschrift Stern in seiner Ausgangsform.



Nun wurde die Kandidatenliste auf Felder mit nur einer möglichen Zahl untersucht und diese ggf. gesetzt.



Jetzt wurde die Kombinierte-Lösungsvariante bestehend aus der Kandidatenliste und André-Technik angewendet.



Am Ende wurde dann noch die Zufalls-Lösungsvariante auf das Rätsel angewendet und somit gelöst.