

Lehrziele:

- Geschachtelte `for` Schleifen
- Strings und Char

Aufgabenstellung

Bei einem Sudoku dürfen in einer Spalte/Zeile/3x3-Zelle nur einmal die Zahlen von 1 bis 9 vorkommen. Versucht man ein Sudoku zu lösen, muss man für die noch freien Zellen bestimmen, welche Zahlen noch möglich sind.

Sind z.B. in einer Zeile die Zahlen 1,2,3,4,5 und 6 (in beliebigen Zellen) enthalten, so darf in den drei noch freien Zellen die Zahlen 7, 8 und 9 geschrieben werden – vorausgesetzt natürlich, man missachtet nicht die gleiche Regel für die Spalte und 3x3-Zelle.

Gesucht ist ein Programm, welches die möglichen Werte aus Sicht einer Zeile/Spalte/3x3-Zelle bestimmt. Der Benutzer gibt dazu die bereits in der Zeile/Spalte/3x3-Zelle enthaltenen Werte an, das Programm berechnet anschließend die noch fehlenden Werte.

Das Programm orientiert sich an dem folgenden Ausgabebeispiel:

```
Please enter all digits in a row/col/3x3 (empty to exit): 1
23456789 is possible
Please enter all digits in a row/col/3x3 (empty to exit): 13579
2468 is possible
Please enter all digits in a row/col/3x3 (empty to exit): 8642
13579 is possible
Please enter all digits in a row/col/3x3 (empty to exit): 123654789
nothing is possible
```

Hinweise:

- Vordefinierte String Funktionen (z.B. `Contains`, `SubString`, ...) dürfen nicht verwendet werden.
- Sie können einen String „123456789“ anlegen der alle gültigen Zeichen(=Zahlen) definiert. Diesen können sie zeichenweise durch gehen um die fehlenden Zeichen(=Zahlen) aus der Usereingabe zu finden.

Zusatzaufgaben

- Überprüfen sie die Eingabe. Geben sie seine Fehlermeldung aus, wenn diese ungültig ist.
 - Gültige Zeichen sind 0..9 und ein „Blank“
 - ‚0‘ bzw. Blank wird ignoriert
 - Ziffern dürfen nicht doppelt vorkommen.Die Eingabe „11“ ist ungültig