

---

## Aufgabe C2: Matrix transponieren

Schreiben Sie ein Programm, das quadratische Integer-Matrizen transponiert, also die Zeilen und Spalten der Matrizen vertauscht, wie folgendes Beispiel illustriert:

$$\begin{pmatrix} 15 & 8 & 1 & 24 & 17 \\ 16 & 14 & 7 & 5 & 23 \\ 22 & 20 & 13 & 6 & 4 \\ 3 & 21 & 19 & 12 & 10 \\ 9 & 2 & 25 & 18 & 11 \end{pmatrix}$$

wird transponiert zu

$$\begin{pmatrix} 15 & 16 & 22 & 3 & 9 \\ 8 & 14 & 20 & 21 & 2 \\ 1 & 7 & 13 & 19 & 25 \\ 24 & 5 & 6 & 12 & 18 \\ 17 & 23 & 4 & 10 & 11 \end{pmatrix}$$

Vertauschen Sie nur die Elemente der Matrix selbst (also *“in place”*), d.h. verwenden Sie keine Hilfsmatrix.

Da es mühselig ist, die Werte der Elemente von Matrizen selbst einzugeben und im Lerntext keine Möglichkeit zum Einlesen aus Dateien eingeführt wird, geben wir in der Datei `gibMatrix.go` eine Funktion vor, die einige Integer-Matrizen<sup>1</sup> erzeugt:

```
// gibtMatrix gibt eine von vier quadratischen 5x5-Integer-Matrizen zurück.  
// nr (1 bis 4) gibt an, welche Beispiel-Matrizen zurückgegeben wird.  
func gibMatrix(nr int) [5][5]int {...}
```

Ihr Programm soll die Funktion mit den Argumenten 1 bis 4 aufrufen, die erhaltenen Matrizen ausgeben, transponieren und wieder ausgeben. Bis auf die Formatierung durch Leerzeichen soll genau die Ausgabe entstehen, die wir als Datei `ausgabe.txt` beifügen. Sie können die Ausgabe Ihres Programmes folgendermaßen in eine Datei umleiten:

```
go run . > meine-ausgabe.txt
```

Nun können Sie die beiden Dateien optisch vergleichen und/oder ein Dateivergleichsprogramm zur Hilfe nehmen. Wenn Sie beispielsweise `diff` verwenden, sollte der folgende Befehl keine Unterschiede auführen:

---

<sup>1</sup>Da Arrays nur statisch mit fester Dimension erzeugt werden können, werden als Beispiele ausschließlich 5x5 Matrizen verwendet.

---

```
diff --ignore-all-space meine-ausgabe.txt ausgabe.txt
```

Im Unterordner ML finden Sie einen Lösungsvorschlag.