
Aufgabe D2: Alternierende Spaltensumme

In dieser Aufgabe geht es um eine Matrix (d.h. ein zweidimensionales Array), die folgende Eigenschaft haben soll:

Jeder Werte in der letzten Zeile der Matrix entspricht der alternierenden Summe der Elemente der jeweiligen Spalte. Beispielsweise hat die folgende Matrix diese Eigenschaft:

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
27	-5	13	31	-1

Denn hier gilt für jeden Wert in der letzten Zeile: Der Wert ergibt sich, indem die Elemente der entsprechenden Spalte abwechselnd addiert und subtrahiert werden. Die folgenden Berechnungen illustrieren dies:

$$\begin{aligned} 27 &= 15 - 16 + 22 - 3 + 9 \\ -5 &= 8 - 14 + 20 - 21 + 2 \\ 13 &= 1 - 7 + 13 - 19 + 25 \\ 31 &= 24 - 5 + 6 - 12 + 18 \\ -1 &= 17 - 23 + 4 - 10 + 11 \end{aligned}$$

Schreiben Sie eine Funktion

```
func alternierendeSpaltensumme(arr *[anzZeilen][anzSpalten]int) int {...}
```

die eine (per Zeiger) übergebene Matrix daraufhin überprüft, ob in der letzten Zeile der Matrix die alternierenden Spaltensummen stehen. Falls dies nicht der Fall ist, sollen die letzte Zeile der Matrix entsprechend korrigiert werden. Die Funktion soll abschließend die Anzahl der fehlerhaften Spaltensummen zurückgeben (also ggf. 0, falls bereits alle korrekt waren).

Die Größe der Matrix (bzw. die Länge der Arrays) ist mittels der Konstanten `anzZeilen` und `anzSpalten` in der Datei `main.go` festgelegt.

Ressourcen

Im Ordner dieser Aufgabe finden Sie eine Datei `alternierendeSpaltensumme.go`, die Sie entsprechend der Aufgabenstellung abändern sollen.

Weiterhin stellen wir in der Datei `main.go` eine `main`-Funktion zur Verfügung, damit Sie Ihre Funktion in einem beispielhaften Kontext kompilieren und ausführen können. Nachdem Sie in den Aufgabenordner gewechselt sind, geben Sie dazu folgenden Befehl ein:

```
go run .
```

Die Datei `alternierendeSpaltensumme_test.go` stellt Tests bereit, die Sie mit folgendem Befehl durchführen können:

```
go test
```

Im Unterordner `ML` finden Sie einen Lösungsvorschlag.