

Programmieren C:

Variabel große zweidimensionale Arrays und Funktionen: Magisches Quadrat

Klaus Kusche

Ein magisches Quadrat der Seitenlänge n ist eine quadratische Anordnung der Zahlen von 1 bis $n*n$, bei dem die Summe der Zahlen in allen Zeilen und Spalten sowie in den beiden langen Diagonalen **gleich** ist.

Für die Berechnung solcher Quadrate gibt es zahlreiche Verfahren, deren Anwendbarkeit u.a. von der Seitenlänge des Quadrates abhängt. Folgendes Verfahren ist relativ einfach, eignet sich aber nur für Seitenlängen, die weder durch 2 noch durch 3 teilbar sind (also 1, 5, 7, 11, 13, ...).

- Man beginnt mit 1 im rechten oberen Eck.
- Für jede nächste Zahl bis n^2 geht man ein Feld nach links und 2 nach unten, nach jeder n-ten Zahl stattdessen waagrecht 2 nach links.
- Man betrachtet das Quadrat dabei in beiden Richtungen als zyklisch (als ob der untere und der obere bzw. der linke und der rechte Rand zylindrisch zusammengeklebt wären):
Kommt man über den unteren Rand, macht man oben weiter,
kommt man über den linken Rand, macht man rechts weiter.

Unser Programm wird mit genau einer ganzen Zahl n auf der Befehlszeile aufgerufen: Der gewünschten Seitenlänge des Quadrates (prüfe, ob eine Zahl angegeben wurde und ob sie größer 0 und weder durch 2 noch durch 3 teilbar ist!).

In der ersten Version soll unser Programm mit einem zweidimensionalen Array arbeiten, das im Hauptprogramm lokal genau in der benötigten Größe $n*n$ deklariert wird. Mach dich daher schlau, wie man ein solches variabel großes zweidimensionales Array laut C11-Standard an Funktionen übergibt (das steht u.a. in meinen Folien).

Schreib zuerst folgende Hilfsfunktionen

(alle haben das Array und seine Größe als Parameter und keinen Returnwert):

- Alle Elemente des Arrays mit derselben übergebenen Zahl befüllen.
- Das Array nach der oben beschriebenen Methode als magisches Quadrat befüllen.
- Alle Elemente des Arrays ausgeben.
- **Optional:** Alle Zeilen, alle Spalten und die beiden langen Diagonalen prüfen, ob deren Summe der Zahlen gleich einem übergebenen Sollwert ist (sonst: Fehlermeldung ausgeben).

Das Hauptprogramm soll

- die Eingabe prüfen und ein Array in dieser Größe deklarieren,
- das Array zuerst mit -9999 und dann mit den Zahlen des magischen Quadrates befüllen,
- das Array ausgeben,
- den Soll-Summenwert $n*(n^2+1)/2$ ausgeben
- und (falls vorhanden) die Prüf-Funktion mit diesem Wert aufrufen.

Dass wir das Array zuerst mit dem fixen Wert -9999 befüllen und erst danach mit den magischen Zahlen, soll uns bei der Fehlersuche helfen: So fallen Elemente, die beim Berechnen des magischen Quadrates versehentlich nicht befüllt werden, in der Ausgabe optisch gleich deutlich auf.

Dynamisch angelegte Arrays

In der zweiten Version wird das quadratische Array dynamisch angelegt, und zwar so, wie es in C seit Jahrzehnten üblich und weit verbreitet ist (und vor C11 auch nötig war): In Form von n einzelnen eindimensionalen Arrays mit den Daten der einzelnen Zeilen (jeweils mit Länge n) und einem weiteren Array mit n Pointern auf diese Zeilen-Arrays.

Schreib dazu eine Funktion, die mit der Seitenlänge n aufgerufen wird, die Arrays dynamisch anlegt (zuerst das für die Pointer und dann die für die Datenzeilen), und einen Pointer auf das Pointer-Array als Returnwert zurückliefert (welchen Typ hat der?). Prüfe dabei alle **malloc**-Aufrufe auf Fehler.

Im Hauptprogramm fällt die bisherige Array-Deklaration weg, stattdessen wird ein Pointer deklariert und durch den Aufruf der Anlege-Funktion initialisiert.

In den anderen Funktionen muss nur die Parameter-Deklaration angepasst werden (ich empfehle die besser lesbare Variante!), der eigentliche Code bleibt völlig unverändert.