

Tag 17 - Clumsy Crucible

- [Variante 1](#)

Lösungshinweise Teil 1

- Die Aufgabe ist ein "Shortest Path Problem" ("Kürzester Pfad Problem") und lässt sich z. B. mit dem [Dijkstra-Algorithmus](#) lösen!
- Die Lösung ist allerdings trotzdem nicht trivial, denn es genügt nicht, die Ziffern in der 2D-Matrix als einzige Knoten aufzufassen. Die Einschränkung, dass Wege maximal 3 Schritte lang in einer Richtung verlaufen dürfen, zwingt einen dazu, jede Koordinate als insgesamt 12 mögliche Knoten aufzufassen. Beispiel zu einer beliebigen Koordinate mitten im Feld: Je nachdem, ob ich die Koordinate von links, rechts, oben oder unten betreten habe und jeweils zuvor 1, 2 oder 3 Schritte lang geradeaus lief, kann die Koordinate unterschiedliche Distanzen aufweisen.
- Ein gutes Beispiel zu diesem Tag, das man auf reddit finden kann, ist folgender kleiner Input:

```
112999
911111
```

Das korrekte Ergebnis muss 7 sein! Dazu muss dein Algorithmus in der Lage sein, auch den zunächst nachteilig erscheinenden Schritt horizontal in die "2" zu betrachten.

Einen Lösungsvorschlag gibt es (von mir) zu diesem Tag allerdings trotzdem nicht, da in Ermangelung von Zeit zwar eine funktionierende Lösung, aber weder eine aufgeräumte noch eine performante

Lösung gefunden wurde 😞

Lösungshinweise Teil 2

Für Teil 2 müssen die Wege-Einschränkungen aus Teil 1 nur minimal überarbeitet werden: jeder Abschnitt muss genau zwischen 4-10 Schritte lang in einer Richtung verlaufen.

From:
<https://www.info-bw.de/> -

Permanent link:
<https://www.info-bw.de/faecher:informatik:oberstufe:java:aoc:aco2023:day17:start?rev=1703009757>

Last update: 19.12.2023 18:15

