

82. Fluten eines Tagbaus, digitale Landschaft, 10 Punkte

TODO: i, j -Werte? Reicht es sich jeweils nur \min für jede Menge zu merken?

Gegeben: Sei S_r die Menge der Seen und I_r die Menge der Inseln ($r \in \mathbb{N}$). Die Höhe des Grundwasserspiegels wird durch h repräsentiert.

Gesucht: Alle Bereiche (Seen) die ab einer Höhe h überflutet werden.

Bei der Vorverarbeitung werden alle Felder $a[i, j]$ aus der Landschaft $L = S_0 \cup \dots \cup S_r \cup I_0 \cup \dots \cup I_r$ mit $h = 0$ entsprechend zu Seen und Inseln zusammengefasst. Ein See S (analog Insel I) ist eine Menge von Feldern, die direkt benachbart (entweder $a[i+1, j]$ oder $a[i, j+1]$, aber nicht $a[i+1, j+1]$) sind. Der Repräsentant der jeweiligen Felder ist die Höhe $h_{a_{i,j}}$. Zu jeder Menge S oder I werden die Koordinaten i, j gespeichert, sodass schneller bestimmt werden kann, ob ein Feld a direkter Nachbar von dieser Menge ist.

FIND(h) liefert das erste Feld $a[i, j] < h$ in L .

UNION(a, S) entfernt a aus der Ursprungsmenge und fügt es zur neuen Menge S hinzu. Die richtige Menge S ist die Menge, wo a ein direkter Nachbar wird. Gibt es zwei Mengen, wo a direkter Nachbar wird, müssen alle Knoten aus der kleineren Menge in die größere überführt werden. (einfacher in Baumstruktur: a wird neue Wurzel).

Algorithmus

Finde alle Felder $< h$ und füge sie der richtigen Menge hinzu.

Laufzeit

Sei $n = |L|$.

FIND(h) $\mathcal{O}(n)$

UNION(a, S) $\mathcal{O}(1)$

Die Ackermannfunktion, 10 Punkte

a) Funktionen

$$A_1(n) = \begin{cases} 3, & n = 1 \\ A_0(A_1(n-1)), & n > 1 \end{cases}$$

$$A_2(n) = \begin{cases} 2, & n = 1 \\ A_1(A_2(n-1)), & n > 1 \end{cases}$$

$$A_3(n) = A_2(A_3(n-1))$$

$$A_4(n) = A_3(A_4(n-1))$$

b) Werte

i	$A_i(1)$	$A_i(2)$
0	2	3
1	3	4
2		
3		
4		