

Aufgabe 2 - Nummernmerker

Team Mineraldifferenz (00694)

Fynn Kiwitt, Erik Klein, Adrian Krauss

Lösungsidee

Das Problem lässt sich mit Dynamischer Programmierung lösen.

Sei $x[0...N-1]$ die Nummer. Sei $M[i]$ die minimale Anzahl von Blöcken, die mit 0 beginnen, wenn man $x[i...N-1]$ in Blöcke aufteilt. Gesucht ist $M[0]$.

Der erste Block ist $x[i...i+1]$, $x[i...i+2]$ oder $x[i...i+3]$. Der übrige Teil der Nummer muss ebenso optimal aufgeteilt werden. Als Rekurrenz:

$$M[i] = \min(M[i+2], M[i+3], M[i+4]) + (x[i] == 0)$$

Die Basisfälle sind: $M[N-1] = \infty$, da ein Block mindestens Länge 2 hat, und für $i = N-4...N-2$ $M[i] = (x[i] == 0)$.

Die Rekurrenz wird von $i = N-5$ bis 0 absteigend evaluiert, sodass $M[i+2]$, $M[i+3]$, und $M[i+4]$ immer schon berechnet wurden, wenn sie für $M[i]$ benötigt werden.

Danach wird eine optimale Lösung durch eine rekursive Funktion rekonstruiert.

Der Algorithmus hat lineare Laufzeit, was asymptotisch optimal ist, da allein das Einlesen der Nummer lineare Zeit benötigt. Er ist effizient, korrekt und einfach zu programmieren. Er kann Zahlen mit Millionen von Stellen innerhalb von Millisekunden bearbeiten!

Umsetzung

Die Idee wurde in C++ umgesetzt. x wird als string der Länge N repräsentiert. Die Rekurrenz ist wie beschrieben umgesetzt.

Die rekursive Funktion `output(i)` zur Rekonstruktion wird auf 0 aufgerufen. Sie bestimmt erst die Länge des Blocks, der bei i beginnt, indem sie bestimmt, welches von $M[i+2]$, $M[i+3]$, und $M[i+4]$ minimal ist. Dann gibt sie den Block mit `string.substr()` aus, und ruft sich auf dem übrigen Teil der Nummer auf.

Beispiele

Die Nummern der BwInf-Website:

Nummer1: 00 54 800 0000 51 79 734 (2 mit 0 beginnende Bloecke)

Nummer2: 03 49 59 29 53 37 90 15 44 12 660 (1 mit 0 beginnende Bloecke)

Nummer3: 53 19 97 48 790 22 72 560 76 20 179 (0 mit 0 beginnende Bloecke)

Nummer4: 90 88 76 10 51 69 94 82 78 90 38 33 1267 (0 mit 0 beginnende Bloecke)

Nummer5: 01 10 000 0001 1000 100 11 11 11 10 1011 (3 mit 0 beginnende Bloecke)

Die ersten 20 Stellen von π : 31 41 59 26 53 58 97 93 2384 (0 mit 0 beginnende Bloecke)

Die ersten 20 Stellen von e : 27 18 28 18 28 45 90 45 2353 (0 mit 0 beginnende Bloecke)

Die ersten 20 Stellen des goldenen Schnitts: 16 180 33 98 87 49 89 48 482 (0 mit 0 beginnende Bloecke)

Die Lichtgeschwindigkeit in m/s: 29 97 92 458 (0 mit 0 beginnende Bloecke)

Quellcode

```
#include <bits/stdc++.h>
using namespace std;

int N, M[1000000000];
string x;

void output(int i) {
    if(i >= N-4) {
        cout << x.substr(i,N-i) << ' ';
        return;
    }
    for(int j : {2,3,4}) if(M[i+j] == min(M[i+2],min(M[i+3],M[i+4]))) {
        cout << x.substr(i,j) << ' ';
        output(i+j);
        return;
    }
}

int main() {
    cin >> x;
    N = x.length();
    M[N-1] = 1000000000; // unendlich
    for(int i = N-4; i <= N-2; i++) M[i] = (x[i] == '0');
    for(int i = N-5; i >= 0; i--) M[i] = (x[i] == '0') + min(M[i+2],min(M[i+3],M[i+4]));
    output(0);
    cout << '\n' << M[0] << " mit 0 beginnende Bloecke.\n";
    return 0;
}
```