

Datenstrukturen und Algorithmen: Übungsblatt #5

Abgabe am 17. Mai 2018

Finn Hess (378104), Jan Knichel (377779), Paul Orschau (381085)

13.5.2018

Aufgabe 1

Aufgabe 2

Aufgabe 3

Aufgabe 4

Teil a)

- 9 / 3 / 5 / 1 / 10 / 4 / 6
- 3 / 9 / 5 / 1 / 10 / 4 / 6
- 3 / 5 / 9 / 1 / 10 / 4 / 6
- 1 / 3 / 5 / 9 / 10 / 4 / 6
- 1 / 3 / 5 / 9 / 10 / 4 / 6
- 1 / 3 / 4 / 5 / 9 / 10 / 6
- 1 / 3 / 4 / 5 / 6 / 9 / 10

Teil b)

Lösung des Dutch Flag Problem:

```
1 void DutchNationalFlag(Color E[], int n) {  
2     int r=0, b=n+1;  
3     int u=1;  
4     while (u<b) {  
5         if (E[u] == rot) {  
6             sawp(E[r+1], E[u]);  
7             r++;  
8             u++;  
9         }  
10        if (E[u] == weiss) {  
11            u++;  
12        }  
13        if (E[u] == blau) {  
14            swap(E[b-1], E[u]);  
15            b--;  
16        }  
17    }  
18 }
```

Der Algorithmus kann ein Array in 3 Kategorien sortieren. Die Idee bei 5 verschiedenen Farben besteht darin, zunächst in die Kategorien "Blau", "Rot" und "Alles Andere" zu sortieren, und dann den Algorithmus ein weiteres Mal auf den Bereich "Alles Andere", in dem dann nur noch 3 verschiedene Farben sind, anzuwenden. Letzteres entspricht einfach dem Dutch National Flag Problem.

Dazu muss der Algorithmus aber leicht abgewandelt werden. Er muss:

- Die dritte Farbe darüber definieren, dass es weder die erste noch die zweite Farbe ist
- Mit verschiedenen Farben, die als Eingabeparameter übergeben werden, arbeiten können
- Die Größe des dritten Bereichs zurückliefern

Hier ist die angepasste Lösung und die beschriebene Verwendung:

```
1  int DutchNationalFlag(Color E[], int n, Color first , Color second) {
2      int r=0, b=n+1;
3      int u=1;
4      while (u<b) {
5          if ((E[u] != first) and (E[u] != second)) { // Element ist "bunt"
6              sawp(E[r+1], E[u]);
7              r++;
8              u++;
9          }
10         if (E[u] == first) { // Element hat Farbe "first"
11             u++;
12         }
13         if (E[u] == second) { // Element hat Farbe "second"
14             swap(E[b-1], E[u]);
15             b--;
16         }
17     }
18     return r; // Anzahl der Elemente im ersten ("bunten") Bereich
19 }
20
21 int main() {
22     // E ist gegeben, 5 Farben, unsortiert
23
24     int x = DutchNationalFlag(E, E.size , GELB, GRUEN); // Lauzeit in Theta(n)
25
26     // E enthaelt im Bereich 1 bis x nur BLAU, ROT und SCHWARZ unsortiert
27     // und im Bereich x+1 bis n nur GELB (links) und GRUEN (rechts)
28
29     DutchNationalFlag(E, x, ROT, SCHWARZ); // Laufzeit in Theta(x)
30
31     // E ist sortiert
32     // BLAU, ROT, SCHWARZ, GELB, GRUEN
33     return 0;
34 }
```

Dieser Algorithmus hat immernoch eine lineare Laufzeit in $\Theta(n)$ und einen konstanten Speicherbedarf.