

# AD Praktikum: Aufgabe 09, Graph and Dykstra

Sönke Peters  
Karl-Fabian Witte

5. Juni 2017

## Abstract

Die Datenstruktur eines gewichteter Graphen wird auf Adjazenzmatrix und -liste implementiert und der Algorithmus von Dykstra wird auf diesen mit einer Komplexitätsuntersuchung ausgewertet.

## 1 Aufgabenstellung

Es sollen zwei Implementationen von dem abstrakten Datentyp Graph realisiert werden. Die Komplexität des Dykstras Algorithmus ist auf beiden Graphenimplementationen zu messen. Die für den Algorithmus wichtigen zusatzinformationen sind nicht im Graphen selbst gespeichert. Zudem sollen zufallsgeneriertere Graphen erstellt werden, welche für die Messung der Komplexität verwedet werden.

### 1.1 Grapheninterface und Implementationsarten

#### 1.1.1 Adjazenzmatrix

#### 1.1.2 Adjazenliste

### 1.2 Graphengenerator

### 1.3 Algorithmus von Dykstra

### 1.4 Zählerintegration

## 2 Messung

In Abbildung 1 zu sehen...

```
Knoten find_min(int m){
    Knoten a_m = 0;
    Knoten tmp = root;
    Knoten last_gt = root;
    while (a_m == 0){
        if(m == tmp.wert){
            a_m = tmp;
        } else {
            if ( m < tmp.wert ){
                last_gt = tmp;
                if (tmp.linkesKind == 0) am = last_gt;
                else tmp = tmp.linkesKind;
            }
        }
    }
}
```

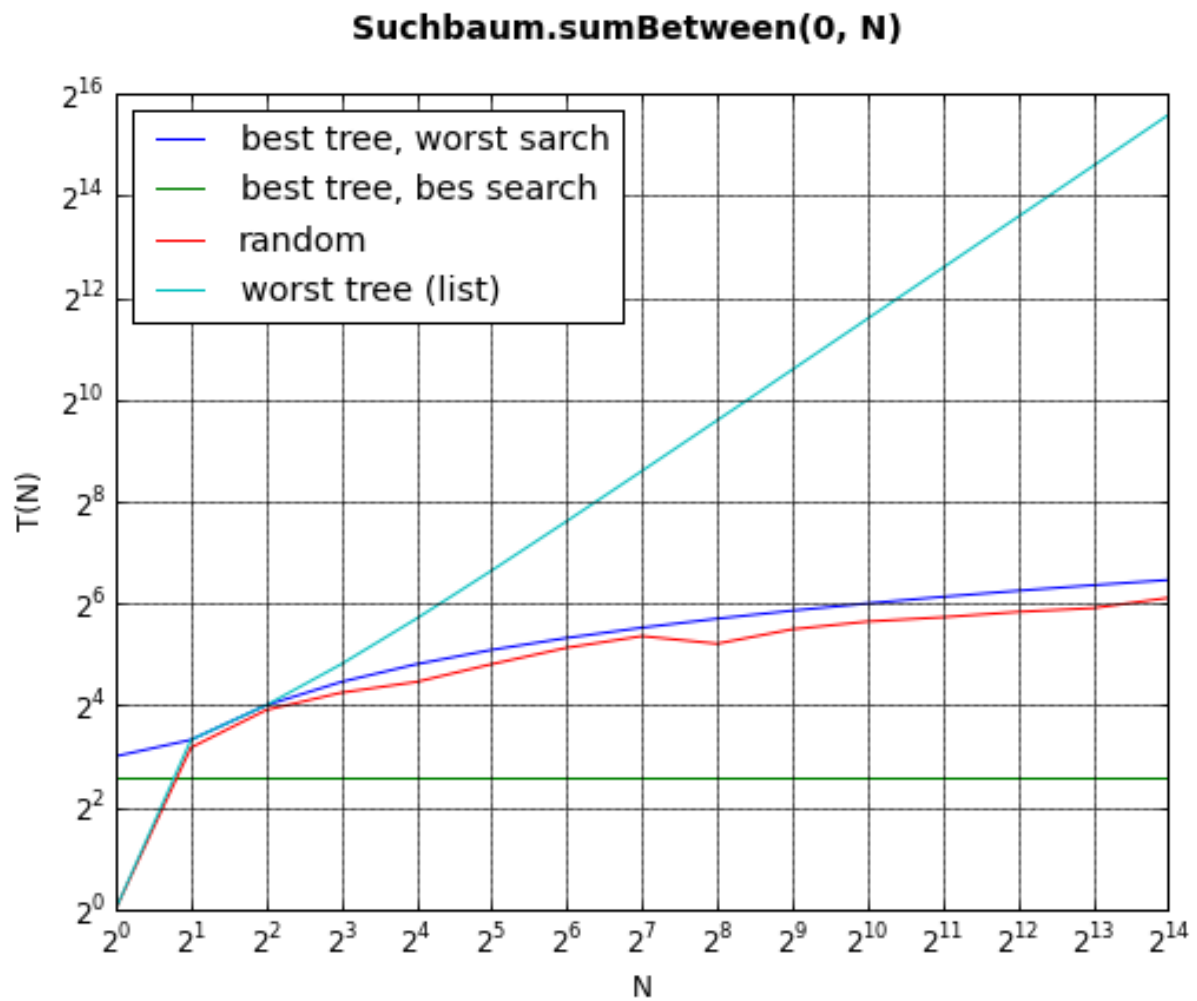


Abbildung 1: Aufwende der Suche in Abhängigkeit von der Baumstruktur und der geuchten Min und Max Werte, sowie der Anzahl der Knoten N

```

    }
    else if (m > tmp.wert ){
        if (tmp.rechtesKind == 0) am = last_gt;
        else tmp = tmp.rechtesKind;
    }
}
return a_m;
}

```