

Übungsblatt Algorithmetik

Vorkurs Informatik WS 2016/2017

Dennis Aumiller

Aumiller@stud.uni-heidelberg.de

12. Oktober 2016

Aufgabe 1

Erinnert euch an die in der Vorlesung vorgestellten Methoden zur Abschätzung der Laufzeit. Der folgende Code beschreibt den ebenfalls bereits vorgestellten Sortieralgorithmus “Bubble-Sort”.

```
void BubbleSort(array L)
{
    n = length(L);
    for i = 1,...,n:
        for j = 0,...,n-i:
            // swapping process
            if L[j] > L[j+1]:
                x = L[j]
                L[j] = L[j+1]
                L[j+1] = x
            end
        end
    end
}
```

1.1 Führt den Algorithmus am Beispiel [11, 9, 12, 18, 17] durch.

1.2.1 Überlegt euch jeweils ein Beispiel für einen optimale sowie schlechtestmöglichen Durchlauf des Algorithmus. Wie müssen die Zahlen dafür jeweils angeordnet sein?

1.2.2 Ordnet die daraus resultierende Laufzeit in die entsprechende Klasse ($\mathcal{O}(\text{irgendwas})$) ein. Was fällt euch dabei auf? Sind die genannten Fälle realistisch, bzw mit welcher Laufzeit kann man in real-world-Anwendungen rechnen?

Aufgabe 2

Entwerft einen KI-Gegner für ein Tic-Tac-Toe-Spiel.

Um das ganze möglichst strukturiert zu halten, empfehle ich, sich folgende Ideen von vornherein zu überlegen: Fängt immer derselbe Spieler an? Falls ja, gibt es eine optimale Strategie als Startzug? Selbstverständlich dürft ihr hierzu Pseudocode wie “X in gegenüberliegende Ecke setzen” verwenden, um es übersichtlich zu halten.

Eine knappe Beschreibung der “perfekten Taktik” gibt es hier oder hier zu finden.

Aufgabe 3

Schreibe eine rekursive Funktion, die die Summe aller Zahlen bis N (natürliche Zahl bildet)