Übungsblatt 6 für "Diskrete und geometrische Algorithmen"

1.) Ein alternativer Ansatz zum Berechnen des Erwartungswertes $\overline{T}_n$ der Anzahl an Vergleichen zwischen Datenelementen (= Schlüsselvergleiche) beim (randomisierten) Quicksort.

(a) Man überlege sich, daß $\overline{T}_n$ der folgenden Rekursion genügt:

$$\overline{T}_n = \frac{1}{n} \sum_{k=1}^{n} \left(\overline{T}_{k-1} + \overline{T}_{n-k} + n - 1\right) = n - 1 + \frac{2}{n} \sum_{k=1}^{n} \overline{T}_{k-1}, \quad n \geq 1, \qquad \overline{T}_0 = 0.$$

(b) Durch Multiplizieren dieser Rekursion mit dem Faktor n und anschließendem Subtrahieren der entsprechenden Gleichung für $\overline{T}_{n-1}$ erhält man eine lineare Rekursion erster Ordnung, welche man löse.

2.) Man zeige mit Hilfe der Substitutionsmethode, daß für die Lösung $T(n)$ der Rekursion

$$T(n) = \max_{1 \leq k \leq n} \big(T(k-1) + T(n-k)\big) + \Theta(n)$$

das asymptotische Verhalten $T(n) = \Omega(n^2)$ gilt. Dabei überlege man sich, daß das Maximum der Funktion $f(n,k) := (k-1)^2 + (n-k)^2$, mit $1 \leq k \leq n$, für festes n immer "am Rand" angenommen wird.

3.) Aus einer gegebenen Menge von n Zahlen wollen wir die i größten in sortierter Reihenfolge bestimmen, wobei wir einen vergleichsbasierten Algorithmus verwenden wollen. Überlegen Sie sich für jeden der folgenden Ansätze jeweils einen Algorithmus mit bestmöglicher asymptotischer Laufzeit im schlechtesten Fall und analysieren Sie die Laufzeiten in Abhängigkeit von n und i.

(a) Sortieren Sie die Zahlen und listen Sie die i größten auf.

(b) Konstruieren Sie aus den Zahlen eine Max-Prioritätswarteschlange und rufen Sie Extract-Max i-mal auf.

(c) Verwenden Sie einen Algorithmus zur Bestimmung der Ranggröße, um die i-größte Zahl zu finden, verwenden Sie diese Zahl als Pivotelement, um die Eingabefolge zu partitionieren und sortieren Sie dann die i größten Zahlen.

Anmerkung: Für (c) dürfen Sie verwenden, daß ein Algorithmus für das Auswahlproblem existiert, welcher dieses im schlechtesten Fall mit $\Theta(n)$ löst.