Übungsblatt 4

25.) (a) Welche Laufzeit benötigt die Berechnung des Produktes einer $(kn) \times n$-Matrix A mit einer $n \times (kn)$-Matrix B, wenn Strassens Algorithmus als Unterprogramm verwendet wird?

(b) Beantworten Sie die gleiche Frage für die Berechnung von $B \cdot A$.

26.) Erzeugt die folgende Modifikation des Fisher-Yates-Algorithmus ebenfalls eine zufällige Permutation von A? Weshalb oder weshalb nicht?

```
PERMUTE-WITH-ALL(A)
n := |A|
for i = 1 to n do
   vertausche A[i] mit A[RANDOM(1, n)]
END
```

27.) Sei $0 \leq m \leq n$. Zeigen Sie, dass die folgende Prozedur eine zufällige Stichprobe der Größe m aus $\{1, 2, \ldots, n\}$ so bestimmt, dass jede m-elementige Teilmenge mit der gleichen Wahrscheinlichkeit auftritt.

```
RANDOM-SAMPLE(m, n)
IF m = 0 THEN
   RETURN ∅
ELSE    S := RANDOM-SAMPLE(m − 1, n − 1)
   i := RANDOM(1, n)
   IF i ∈ S THEN
      S := S ∪ {n}
   ELSE
      S := S ∪ {i}
   END IF
   RETURN S
END IF
```

28.) Erzeugen eines Heaps durch Einfügen, also durch wiederholten Aufruf der Prozedur MAX-HEAP-INSERT. Betrachten Sie die wie folgt geänderte BUILD-MAX-HEAP-Prozedur:

```
BUILD-MAX-HEAP'(A)
heap-size := 1
for i = 2 to |A| do
   MAX-HEAP-INSERT(A, A[i])
end
```

(a) Beweisen Sie, dass die Prozeduren BUILD-MAX-HEAP und BUILD-MAX-HEAP' immer den selben Heap erzeugen, wenn sie auf das gleiche Eingabefeld angewendet werden, oder geben Sie ein Gegenbeispiel an.

(b) Zeigen Sie, dass Build-Max-Heap' im schlechtesten Fall die Laufzeit $\Theta(n \log n)$ benötigt, um einen Heap mit n Elementen zu erzeugen.

29.) Alternativer Ansatz zum Berechnen des Erwartungswertes $\overline{T}_n$ der Anzahl an Schlüsselvergleichen zwischen Datenelementen beim randomisierten Quicksort:

(a) Zeigen Sie, dass $\overline{T}_n$ der folgenden Rekursion genügt:

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

(b) Bestimmen Sie $n\overline{T}_n - (n-1)\overline{T}_{n-1}$ und lösen die daraus gewonnen lineare Rekursion erster Ordnung.

30.) Bestimmen Sie eine Rekursion für die mittlere Anzahl der Schlüsselvergleiche beim (nicht-randomisierten) Quicksort, wenn anstelle des letzten Elements $A[n]$ das der Größe nach mittlere der drei Elemente $A[1]$, $A[\lfloor \frac{n}{2} \rfloor]$ und $A[n]$ als Pivot-Element verwendet wird. Das Lösen dieser Rekursion ist nicht verlangt.

Bemerkung: Mit Hilfe von *erzeugenden Funktionen* kann diese Rekursion in eine Eulersche Differentialgleichung umgeschrieben werden, die man explizit lösen kann.

31.) 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.

32.) Welche der Sortieralgorithmen Insertion-Sort, Merge-Sort, Heapsort und Quicksort sind stabil?