 Prof. Christian Cachin

Randomisierte Algorithmen – Übungen

1. Von Monte Carlo nach Las Vegas

Gegeben sei ein Monte-Carlo-Algorithmus M für ein Problem Π, welcher für jede Instanz des Problems durchschnittlich T Schritte benötigt und dabei eine korrekte Antwort mit Wahrscheinlichkeit α liefert. Angenommen für Π existiere ein deterministischer Verifikationsalgorithmus V, der in K Schritten feststellt, ob eine mögliche Lösung stimmt. Transformieren Sie M in einen Las-Vegas-Algorithmus L, welcher immer eine richtige Lösung für Π liefert, und zeigen Sie, dass die erwartete Laufzeit $(T + K)/\alpha$ Schritte ist.

2. Zufallserzeugung

Gegeben sei eine Methode *RBIT()*, welche bei jedem Aufruf ein zufälliges, uniform verteiles Bit $R \in \{0, 1\}$ liefert. Konstruieren Sie daraus eine Methode (in Pseudocode), welche Zufallswerte mit einer anderen Verteilung erzeugt, sofern die Methode terminiert. Es darf sein, dass Ihre Lösung mit kleiner Wahrscheinlichkeit (WSK) nicht terminiert.

a) Beschreiben Sie eine Methode *Alpha()*, welche ein Bit $A \in \{0, 1\}$ erzeugt, so dass $A = 1$ mit WSK $\frac{5}{8}$ und $A = 0$ sonst.

b) Beschreiben Sie eine Methode *Beta()*, welche $B \in \{1, 2, 3\}$ erzeugt, so dass $B = 1$ mit WSK $\frac{1}{13}$, $B = 2$ mit WSK $\frac{4}{13}$ und $B = 3$ mit WSK $\frac{8}{13}$.

Beachten Sie dabei (nochmals!), dass die WSK-Verteilung nur in den Fällen gelten muss, dass der Algorithmus auch terminiert hat.