07.06.2002

8. Übung zur Vorlesung Informatik B

Institut für Informatik, FU Berlin, SoSe 2002

1. (2 Punkte) Eine Nachricht sei eine große Primzahl p. Zur Verschlüsselung wird p mit einer anderen geheimen Primzahl q multipliziert. Das Ergebnis $r = p \cdot q$ wird über einen Kanal geschickt und dort vom 'Gegner' mitgelesen. Da der q nicht kennt, decodiert er mit brute force.
   Für $1 < p < r$ teste ob p die Zahl r teilt. Falls ja, schreibe 'Die Nachricht ist p oder r/p', ansonsten mache weiter.

   (a) Wie lange dauert im worst case dieses Verfahren, wenn r 100 Bit hat und eine Division bei Integers bis 100 Bit 10^{-6} Sekunden dauert.

   (b) Geben Sie die worst-case-Abschätzung gemessen in Anzahl der Bytes von r, also $(\log r)/8$. Eine Division soll $O(N)$ kosten.

2. (3 Punkte) Schreiben Sie ein Java Programm zur Bestimmung der Ausgabe 2^i für eine ganzzahlige positive Eingabe i. Ihr Algorithmus sollte nicht i mal die Zahl 2 mit sich selbst multiplizieren, es geht viel schneller.

3. (6 Punkte) Ein Programmtext T kann neben anderen Zeichen Klammern der Form $\{, \}, [,], (,)$ enthalten.
   Wenn S ein leerer String oder ein String ohne Klammern ist, so heißt er korrekt geklammert. Wenn P, P', P'' korrekt geklammert sind, so auch

   $$P'(P)P'', P'\{P\}P'', P'[P]P''$$

   Schreiben und implementieren Sie einen Algorithmus, der unter Benutzung eines Stacks in Zeit proportional zur Länge von T testet, ob T korrekt geklammert ist.

4. (6 Punkte) Entwerfen Sie einen ADT für einen 2-Farben-Doppelstack, also einen roten und einen blauen Stack mit jeweils farbigen Operationen (rotes push,blaues push usw.). Geben Sie eine effiziente Implementierung mittels eines Arrays an. Dessen Größe soll so groß sein wie die der beiden Stacks zusammen.

Achtung: Der geplante Kurztest findet am 12.06. in der Vorlesung statt!

Abgabe: 19.06.2002, 12 Uhr s.t.