Exercise 2.1  Recurrence Relations.

Find a closed form for recurrence relations of the form

\[ T(n) = \begin{cases} 
  aT(n/b) + cn + d & \text{if } n > 1 \\
  e & \text{if } n = 1 
\end{cases} \]

with \(a, b, c, d, e \in \mathbb{N}\), \(b > 1\). Prove your answer using mathematical induction. You can assume that \(n\) is a power of \(b\).

Note: In your proof, distinguish the cases i) \(a \neq b, a \neq 1\) ii) \(a \neq b, a = 1\) and iii) \(a = b\).


Specify (as concisely as possible) the asymptotic running time of the following code fragments in \(\Theta\) notation depending on \(n \in \mathbb{N}\). You do not need to justify your answer.

```java
1    for(int i = 1; i <= n; i += 3) {
2        for(int j = n; j > 1; j = j/3)
3            ;
4        int k = 1;
5        while(k*k <= n)
6            k = k + 2;
7    }

1    for(int i = n; i > 0; i -= 1) {
2        for(int j = 0; j < i; j += 1) {
3            ;
4        }
5    }

1    for(int i = n; i > 0; i -= 1) {
2        for(int j = n; j > 1; j /= 2) {
3            ;
4        }
5    }
```

Please turn over.
Exercise 2.3  Blum’s Median-of-Medians Strategy.

We consider finding the median of a sequence using the median-of-medians strategy from the lecture (see Chapter 3.1 in the book). We will consider only the highest level of recursion, so only the very first invocation of the procedure \texttt{Auswahl} that determines the \(i\)-th smallest element with \(i = \lceil \frac{N}{2} \rceil\).

a) Given the following sequence

\[4, 14, 7, 6, 16, 3, 15, 8, 27, 17, 13, 11, 26, 9, 21, 1, 7, 10, 2, 5, 26, 12, 22, 19, 18\]

provide the two sequences on which \texttt{Auswahl} invokes itself recursively.

b) In general, how long are each of the two sequences used in the two recursive calls of the procedure \texttt{Auswahl} for \(i = \lceil \frac{N}{2} \rceil\) at least and at most?

\textit{Hint:} For the analysis you may assume that all numbers in the input sequence are pairwise distinct.

Exercise 2.4  Open Hashing.

a) Insert the keys 3, 11, 8, 12, 10, 4 in this order into an initially empty hash table of size 7. Use open addressing with the hash function \(h(k) = k \mod 7\) and resolve the conflicts using

(i) linear probing

(ii) quadratic probing, and

(iii) double hashing with \(h'(k) = 1 + (k \mod 5)\).

For each method, provide the number of collisions. Which one is best for the above situation?

b) Which problem occurs if the key 3 is removed from the hash tables in a), and how can you resolve it? Which problems occur if many keys are removed from a hash table?

c) Decide which of the following functions can be used as a second hash function for double hashing, and which ones cannot. Think about the requirements that such a function should fulfil, and justify your answers.

The hash function is \(h(k) = k \mod p\) where \(p\) is a prime, the hash table has \(p\) entries and \(q\) is the largest prime smaller than \(p\). The complete hash function in the \(j\)-th step is \((h(k) - j \cdot h'(k)) \mod p\) and \((h(k) - s(j, k)) \mod p\), respectively.

1. \(h'(k) = (k \cdot p + 1) \mod p\)
2. \(h'(k) = \lceil \ln(k) \rceil \mod q\)
3. \(s(j, k) = ((k + j) \mod p) + 1\)
4. \(s(j, k) = ((k \cdot j) \mod q) + 1\)

d) Which advantage does double hashing have when you compare it to quadratic probing?

\textbf{Hand-in:} until Wednesday, 4th March 2015.