Universität Innsbruck - Institut für Informatik
Prof. Clemens Ballarin, Robert Binna, Friedrich Neurauter, Francois Scharffe and Sarah Winkler

3 June 2008

Proseminar Algorithmen und Datenstrukturen

Exercise Sheet 10

Exercise 1 (Master Theorem)

Give a recurrence equation for the worst-case running time and a tight asymptotic (Θ-notation) bound on the worst-case running time of the following algorithms:

a) Binary search in a sorted array of size n.

b) $StoogeSort(A, 1, n)$ where n is the size of the array A.

Listing 1 *StoogeSort*

Input: array A, start index i, stop index j
Output: the array elements $A[i..j]$ are sorted!

if $A[i] > A[j]$ **then**
    exchange $A[i] \leftrightarrow A[j]$
if $i + 1 \geq j$ **then**
    return
$k := \lfloor (j - i + 1)/3 \rfloor$
$StoogeSort(A, i, j - k)$
$StoogeSort(A, i + k, j)$
$StoogeSort(A, i, j - k)$

Exercise 2 (Stooge Sort)

Prove that $StoogeSort(A, 1, length(A))$ correctly sorts the input array A by induction on the length of A.

Exercise 3 (Quicksort)

Construct a non-trivial example for which quicksort will use $\Omega(n^2)$ comparisons when the pivot is chosen by taking the median of the first, last and middle elements of the sequence (the median of a finite list of numbers can be found by arranging them from lowest value to highest value and picking the middle one, e.g. $median(4, 3, 5) = 4$).

Exercise 4 (Sorting Algorithms)

a) Implement *mergesort* in C and incorporate it into the framework of Exercise 3 of last week.

b) Implement *quicksort* in C and incorporate it into the framework of Exercise 3 of last week.