Data Structures and Algorithms
Exam

Prof. Dr. M. Böhlen

July 10, 2008
14:00 - 16:00

As auxiliary material you may use 1 A4 sheet with your notes. The exam consists of 4 exercises. The four exercises have equal weight. It is important that you argue for your answers and that you present your solutions in a readable form. Write clearly in terms of math, language, and legibility. The clarity of your explanations impacts your grade. The exam lasts 2 hours. Write your name and ID on each solution sheet. All the best.

1 Sorting

Heapsort is a comparison-based sorting algorithm that uses a heap to sort an array in asymptotically optimal running time. Assume we use a MinHeap to sort an array in descending order.

1. Is array A1 = | 2 | 5 | 3 | 8 | 6 | 4 | 22 | 11 | a MinHeap? Explain your answer.
2. Consider the following code fragment:

```
HeapsortDesc(A, n)
  BuildMinHeap(A, n)
  for i := n to 2 do
    Exchange(A[1], A[i])
    n := n-1
    MinHeapify(A, 1, n)
```

   Give the pseudo code for the functions *BuildMinHeap* and *MinHeapify* that are used to sort an array in descending order.
3. Consider array A2 = | 4 | 1 | 3 | 2 | 16 | 9 | 10 | and the call *HeapsortDesc(A2,7)*. Show the content of A2 at the beginning of each iteration of the for loop in *HeapsortDesc*.
4. A sorting algorithm is *stable* if it leaves the order of equal elements unchanged. Is *HeapSortDesc* a stable sorting algorithm? Explain your answer.

2 Red-Black Tree

Consider the Red-Black tree in Figure 1. Circled nodes 24, 8, 20, 28, 44 are black. Rectangle nodes 16, 32, 12, 38 are red. Perform the following operations on the tree in Figure 1. Draw the resulting tree after each operation. For each operation use the orginal tree in Figure 1. Write down the cases you applied.

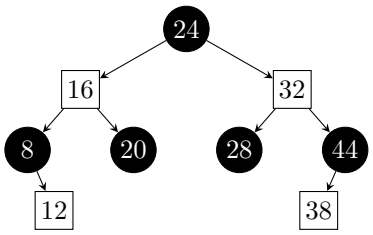

Figure 1: Red-Black Tree

1. Insert 40
2. Delete 20
3. Insert 13
4. Delete 32

3 Arbitrage

In economics arbitrage is the practice of taking advantage of a price differential between two or more markets: a combination of matching deals are struck that capitalize upon the imbalance, the profit being the difference between the market prices. An example arbitrage is to use exchange rates to transform one unit of a currency into more than one unit of the same currency.

Consider the exchange rates in Table 1. For example with the given exchange

	USD	GBP	CAD	EUR	AUD
USD	1	1.98	0.98	1.57	0.96
GBP	0.50	1	0.49	0.79	0.48
CAD	1.02	2.02	1	1.60	0.98
EUR	0.64	1.26	0.62	1	0.61
AUD	1.04	2.06	1.02	1.64	1

Table 1: Table with Exchange Rates

rates one can exchange a USD into GBPs, the GBPs into Euros and the Euros back into USDs. For 1 USD you would get 0.50 * 1.26 * 1.57 = 0.989 USD.

1. Model the above arbitrage problem as a general graph problem. Note that $A * B > 1$ can be rewritten as $(1/A) * (1/B) < 1$, which can be rewritten as $\log(1/A) + \log(1/B) < 0$.
2. Give an efficient algorithm to determine whether or not there exist a sequence of currencies $c_1, ..., c_k$ such that $xch[c_1, c_2] * ... * xch[c_{k-1}, c_k] * xch[c_k, c_1] > 1$.
3. Determine the algorithmic complexity of your solution.

4 String Edit Distance

The *edit distance* between source string x and target string y is the cost of the least expensive sequence of operations that transforms string x into string y.

Permitted operations to transform x into y are

- to copy a character from x to a matching character in y,
- to swap two characters in x and copy them to adjacent matching characters in y,
- to replace a character from x by a character in y,
- to delete a character from x, or
- to insert a character into y.

1. Develop a recursive algorithms that computes the edit distance between a source string x and a target string y.
2. Use dynamic programming to develop an efficient algorithm to compute the edit distance between two strings. Describe the data structure you use for your solution and illustrate it for strings $x =$ "house" and $y =$ "duos".
3. Assume the following costs are given: replace (4), insert (1), and delete (2). Discuss the implications of these weights.