Carina Edlund, CSC, KTH

Written exam in the course

DD2325 Applied Programming and Computer Science

December, 16 2009 kl. 14 - 18

No means of help allowed.
This exam is for the grades D (11 p needed), C (15 p needed), B (19 p needed) and A (22 p needed). When you have passed the mandatory labbs you have passed the course with the grade E.

1 2^n (4 p)

2^n may be defined by the algorithm *twoPower*:

$$twoPower(n) = \begin{cases} 1, & n = 0 \\ 2 * twoPower(n-1) & otherwise \end{cases}$$

where $n = 0, 1, 2, 3 \ldots$

- Write a recursive function in Matlab for calculating 2^n using the algorithm above.
- The pseudocode below, $twoPowerTail$, describes a so called tailrecursive alternative. Show and explain how the function works and what happens when the function call $twoPowerTail(4, 1)$ is performed.

```
function twoPowerTail (n : Integer; result : Integer)
if (n > 0)
   twoPowerTail (n-1 , result*2)
else
   print result
end {*function twoPowerTail*}
```

- Why can the tailrecursive function be more efficient than the recursive one?
- Write a function in C, calculating 2^n, which is iterative.

2 Sending birthday cards (10 p)

In an unsorted *register* there are n structures. Each structure contains the following information on a person; date of birth (month-day), name and address. The sole purpose of the register is to keep track of birthdays and it's only use is when sending birthday cards.

NB! In this text date of birth is month and day. I'm born on the 4'th of june so in this text my date of birth is 0604.

The function `searchStruct` is written to search the *register* and the *currentBirthdate* could be my date of birth 0604.

```
function result = searchStruct(register, currentBirthdate)
found = 0; index = 1;
len = length(register);
data = [];
while (~found) && (index <= len)
    if currentBirthdate == register(index).birthdate
        found = 1
        result = index;
    else
        index = index + 1
    end %if
end %while
end %searchStruct
```

- How many structures needs to be searched in the worst case when using `searchStruct`? Motivate your answer!
- I want the indices for all persons born on a specific day and in a specific month. Explain and motivate why the function `searchStruct` can or can not give this information. If the function fails to return all indices, correct the function `searchStruct` so that it returns all indices.
- Discuss different sorting methods to sort the *register*, when to use them and why. Motivate your answer carefully.
- The sorted *register* is transfered to a C-programme. How would you store the *register*, which datastructures would you use? Explain carefully, make sketches and/or give type definitions in C.
- For efficiency reasons the search is performed with binary search instead. Why is it considered more efficient than the previous? Write an algorithm for binary search. Use your datastructure in previous question.

3 Goal (4 p)

One of the goals of this course is:
"After completing the course the student should be able to:
describe fundamental algorithms for compression"

Motivate in a suitable way and so that it is beyond any doubt that you have reached this goal when it comes to Huffman coding.

4 Binary tree (7 p)

In these questions you only need to consider binary trees.

- Explain and show with an example what is meant by `root node`, `left subtree`, `children` and `leaf node`.
- What are the differences between a `queue` and a `tree`?
- Write a function in C that determines the depth of a tree. If you have to make assumptions you must clearly state and carefully explain them.