Exam (example)

Complexity 2011-2012

A. Silva, H. Barendregt, B. Westerbaan & B. Westerbaan

Exercise 1. (30 points) Consider the following algorithm which produces an array of size n containing the factorial of i in position i.

```
 void factorials (int a[], int n) {
        int i;
        for (i=0; (i<n); i++) a[i] = factorial (i);
     }

 int factorial(int n) {
        int i, k;
        k = 1;
        i = 0;
        while (i < n) {
             i++;
             k= k*i;
        }
 return k;
}
```

(a) **(15 points)** Write an expression for the time execution function $T(n)$ of the algorithm. Characterize the asymptotical behavior of the algorithm using Θ notation.

(b) **(15 points)** Prove that indeed your $T(n)$ is in the Θ class you answered in the previous question (state what you need to prove; give explicitly all the constants involved).

Exercise 2. (20 points) Assume you have functions f and g such that $f(n)$ is in $O(g(n))$. For each of the following statements, decide whether you think it is true or false and give a proof or a counter-example.

1. $\log_2 f(n)$ is $O(\log_2 g(n))$
2. $2^{f(n)}$ is $O(2^{g(n)})$
3. $f(n)^2$ is $O(g(n)^2)$

Exercise 3. (20 points) Consider the following algorithm which returns the minimum element in a binary search tree (with only positive numbers, -1 means the tree was empty).

```
int min(Tree t) {
int x = t-> node;

  if (t) { // if tree is not empty
    while (t->left) // while left branch has elements
         t = t->left;
         x = t->node;
         return x;
         }
         else return -1;
         }
```

In what follows, give your answer using O notation.

(i) **(10 points)** What is the worst case running time for this sorting algorithm? Explain your answer.

(ii) **(10 points)** What is the best case running time for this sorting algorithm? Explain your answer.

Exercise 4. (30 points) Consider the following recursive function which calculates the balancing factors of a tree. Assume the tree a is balanced and that `height` executes in linear time.

```
void bals(Tree a) {
        if (!a) return;

        a->bal = height(a->right) - height(a->left);
         % a->right(left) gives the right(left) subtree of a.
        bals(a->left);
        bals(a->right);
}
```

(a) **(10 points)** Write the execution time function T of the algorithm above (in the form of a recurrence).

(b) **(10 points)** Draw the recursion tree of the algorithm and determine how many levels the tree will have, the cost per level, and the total cost of the algorithm (if convenient, you can overestimate the costs). Guess the asymptotical behavior of the algorithm using O notation.

(c) **(10 points)** Prove that indeed your T is in the O class you answered in the previous question (again: state what you need to prove; give explicitly all the constants involved).