University of Zurich

Department of Informatics
Database Technology Group
Prof. Dr. M. Böhlen

# Informatics II
# Exercise 3

### Week 04

## Algorithmic Complexity

**Task 1.** Below is a pseudocode of a function named `whatDoesItDo`, which takes an array `A[1..n]` of $n$ integers and an integer $k$ as inputs.

---

**Algo:** `whatDoesItDo`(A, n, k)

---

$result = -1000$
**for** $i = 1$ **to** $n$ **do**
    $current = 0$
    **for** $j = i$ **to** $n$ **by** $k$ **do**
        $current = current + A[j]$
    **if** $current > result$ **then**
        $result = current$
**return** $result$

---

*Note:* In the above pseudocode, **for** $j = i$ **to** $n$ **by** $k$ means we do not increase $j$ by 1, but each time, we increase it by $k$, i.e., `j=j+k`.

a) Assume `A[1..n]=[1,3,5,4,2,6,8]` and $k = 3$, describe how the function works step by step and calculate what should be returned by this function.

b) Describe what the algorithm does and what the output is in general cases.

c) Perform exact analysis of the running time of the algorithm.

d) Determine the asymptotic complexity of the algorithm?

e) Implement `whatDoesItDo` in C.

## Asymptotic Complexity

**Task 2.** Calculate the asymptotic tight bound for the following functions and rank them by their order of growth (lowest first). Clearly work out the calculation step by step in your solution.

University of Zurich

Department of Informatics
Database Technology Group
Prof. Dr. M. Böhlen

$$f_1(n) = (2n + 3)!$$
$$f_2(n) = 2\log(6^{\log n^2}) + \log(\pi n^2) + n^3$$
$$f_3(n) = 4^{log_2 n}$$
$$f_4(n) = 12\sqrt{n} + 10^{223} + \log 5^n$$
$$f_5(n) = 10^{\lg 20} n^4 + 8^{229} n^3 + 20^{231} n^2 + 128 n \log n$$
$$f_6(n) = \log n^{2n+1}$$
$$f_7(n) = 101^{\sqrt{n}}$$
$$f_8(n) = \log^2(n) + 50\sqrt{n} + \log(n)$$
$$f_9(n) = n^n + 2^{2n} + 13^{124}$$
$$f_{10}(n) = 14400$$

# Special Case and Correctness Analysis

**Task 3.** Consider the algorithm `algo1`. The input parameters are an array `A[1..n]` with n distinct integers and $k \leq n$.

```
Algo: algo1(A, n, k)

sum = 0;
for i = 1 to k do
    maxi = i;
    for j = i to n do
        if A[j] > A[maxi] then
            maxi = j;
    sum = sum + A[maxi];
    swp = A[i];
    A[i] = A[maxi];
    A[maxi] = swp;
return sum
```

a) Specify Specify the pre/post conditions of the `algo1` algorithm.

b) For the two `for` loops in the algorithm:

    i. Determine if the loop is `up` loop or `down` loop.

    ii. Determine the invariants of these two loops and verify whether they are hold in three stages: `initialization`, `maintenance` and `termination`.

c) Identify some edges cases of the algorithm and verify if the algorithm has the correct output.

d) Conduct an exact analysis of the running time of algorithm `algo1`.

e) Determine the best and the worst case of the algorithm. What is the running time and asymptotic complexity in each case?

University of Zurich

Department of Informatics
Database Technology Group
Prof. Dr. M. Böhlen

# Tasks in past exams

[2021 Final Exam] Assume $f_1(n) = O(1)$, $f_2(n) = O(N^2)$, and $f_3(n) = O(N \log N)$. From these complexities it follows that $f_1(n) + f_2(n) + f_3(n) = O(N \log N)$.

Answer: ☐ True ☐ False