TECHNISCHE UNIVERSITÄT
CHEMNITZ

Department of Computer Science
Chair for Practical Computer Science
Prof. Dr. G. Rünger
Dr. J. Lang, Th. Jakobs, M.Sc.

# 4th Task Sheet

## Task 1

On the webpage to the lecture, you will find the file `linked_list.c`, which is a template for a data structure `List`. This data structure implements a linked list with integer elements whose elements are sorted in ascending order. The list may contain each number at maximum once. The following operations are possible:

- Inserting an element into the list. If the number is already present in the list, give a feed-back.
- Removing an element from the list.
- Finding an element in the list. If the element is present, return 1, otherwise 0.

Modify the source code such that accessing the list in parallel becomes possible. Hence, the access to the list has to be synchronised.

Provide two implementations. Use

1. OpenMP-Lock variables and
2. Critical sections

for synchronisation.

Evaluate the performance of your implementation. For the performance evaluation, use a pre-filled list with 1000 elements. Evaluate the performance by executing a loop $10^6$ times. Each loop iteration should

- insert an element $X$,
- find that element $X$ and
- delete that element $X$.

$X$ is a random integer value calculated for each iteration. Provide the average number of loop iterations per second that are possible with your parallel implementation.