

HOMEWORK 1: BUBBLE SORT

(5 POINTS)

1. Consider the following Bubble sort OpenMP parallelisation:

(2 points)

```
do {
    getauscht = 0;
    #pragma omp parallel for
    for(i = 0; i < MAX-1; i++)
        if(feld[i] > feld[i+1]) {
            temp = feld[i];
            feld[i] = feld[i+1];
            feld[i+1] = temp;
            getauscht = 1;
        }
} while(getauscht);
```

Check if the sorted array of this parallelization is correct. Explain the results.

2. Parallelize sequential Bubble sort algorithm using OpenMP.

(3 points)

HOMEWORK 2: COUNTING SORT

(5 POINTS)

Parallelize the sequential Counting sort algorithm using OpenMP.

HOMEWORK 3: BUCKET SORT

(5 POINTS)

Parallelize the sequential Bucket sort algorithm using OpenMP.

HOMEWORK 4: SELECTION SORT

(5 POINTS)

Parallelize the sequential Selection sort algorithm using OpenMP.

HOMEWORK 5: QUICK SORT

(5 POINTS)

Parallelize the sequential Quick sort algorithm using OpenMP. Hint: use parallel sections to parallelise the recursive calls.

HOMEWORK 6: PERFORMANCE ANALYSIS

(5 · 2 POINTS)

1. Choose a large problem size (i.e. matrix size, graph nodes, highest number) and execute each algorithm for 1, 2, 4, and 8 threads;
2. Compute the speedup and efficiency for each algorithm.