Sarah Naja
Felix Becker
Philipp Hertkorn
Yunus Tan

EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

Programming in C++

Sheet 2

Submission date: 02.10.2024 12:00

In this exercise sheet, you'll implement two sorting algorithms, selection sort and insertion sort. Please use the provided **selection_sort.cpp** and **insertion_sort.cpp** files for your implementation. You'll find a printArray function which outputs the array for your own testing purposes. In addition, we added code to output the time required for sorting so you can perform some basic benchmarking of your implementation (not graded).

2.1 Selection Sort (50 Points)

C++

Sort a given array of integers in ascending order using selection sort and measure its sorting time.

a) Implement the function `selectionSort(int arr[], int nElements)` which takes an integer array and an integer with its number of elements as inputs. Sort the array in ascending order using the selection sort algorithm (see e.g., https://en.wikipedia.org/wiki/Selection_sort). For this, the element at position `i` is swapped with the minimum of the remaining sublist from `i` to `n`, starting from index `i=0` up to index `n`. Use `std::swap(int a, int b)` to swap entries in the array.

b) Test your implementation with the given **testSelectionSort()** function.

2.2 Insertion Sort (50 Points)

C++

Sort a given array of integers in ascending order using insertion sort and measure its sorting time.

a) Implement the function `insertionSort(int arr[], int nElements)` which takes an integer array and an integer with its number of elements as inputs. Sort the array in ascending order using the insertion sort algorithm (see e.g., https://en.wikipedia.org/wiki/Insertion_sort). In insertion sort, one starts with a sorted list of length 1. New elements are added at the end of the list and moved towards the front one step at a time as long as the element to the left has a larger value than the new element. Again, use `std::swap(int a, int b)` to swap entries in the array.

b) Test your implementation with the given **testInsertionSort()** function.