

5. Übung: Sortieren

Ziel: Analyse von Sortierverfahren

1. Aufgabe: Sortierverfahren verwenden

Realisieren Sie die Sortierverfahren Bubblesort, Insertionsort, Selectinsort und Quicksort als Templates, so dass jedes der Verfahren mit unterschiedlichen Typen verwendbar ist. Im Hauptprogramm ist ein Dialog zu entwickeln, der es erlaubt, dass ein Benutzer eine Zeichenkette eingeben und ein Verfahren auswählen kann, das dann verwendet wird, um die Zeichenkette zu sortieren und sortiert auszugeben.

2. Aufgabe: Sortierverfahren analysieren

Erzeugen Sie je ein ein int-Array mit 1.000, 10.000, 100.000 und 1.000.000 Werten unter Verwendung des Zufallszahlengenerators (vgl. Vorlesung). Verwenden Sie die Sortiertemplates aus Aufgabe 1 um für jedes Verfahren zu ermitteln, wie lange es braucht, um je ein Array zu sortieren. Geben Sie die Messwerte am Ende aus, etwa in der Form:

	Bubblesort	Insertion	Selection	Quicksort
1.000:	? sec	? sec	? sec	? sec
10.000:
100.000:
1.000.000:

Hinweis :

Die Zeit, die ein Aufruf von xSort benötigt wird, kann man wie folgt messen:

```
#include <time.h>
...
time_t vorher=time(NULL); // Anzahl Sekunden seit 1.1.1970 0:0
// Sortieren der Daten
xSort(daten);
time_t nachher=time(NULL); // Anzahl Sekunden seit 1.1.1970 0:0

// Ausgeben
cout << "Dauer: " << nachher-vorher << "sec" << endl;
```