



## Programmierpraktikum Technische Informatik (C++)

### Programmiertutorium Vorlesung 7

Pullen Sie von ppti-common. Sie finden dann in ihrem Repository den Ordner `tutorials/Lecture07/`, in dem die folgenden Aufgaben zu bearbeiten sind.

#### Teilaufgabe 1

Vervollständigen Sie die Funktion `exploreVector` in `main.cpp`! Sie soll untersuchen, wie sich der Speicherbedarf eines Vectors von `int` verändert, wenn nacheinander insgesamt `n` Elemente angehängt werden. Die Funktion soll immer dann, wenn sich die `capacity` ändert, die Anzahl der bisher angehängten Elemente, die `size` und die `capacity` ausgeben. Erklären Sie das Verhalten!

#### Teilaufgabe 2

Schauen Sie sich die Funktion `loopVector` in `main.cpp` an!

- a) Was macht die Funktion?
- b) Die Funktion würde nicht funktionieren, wenn der Containertyp `list` oder `forward_list` statt `std::vector` wäre. Warum ist dies so?
- c) Schreiben Sie eine Templatefunktion `loopContainer`, die die Funktionalität von `loopVector` für einen wählbaren sequentiellen Containertyp realisiert! Als Containertypparameter sollen `vector`, `deque` und `list` unterstützt werden.
- d) Warum funktioniert die `loopContainer` nicht für `forward_list` oder `array`?

#### Teilaufgabe 3

Schreiben Sie eine Funktion `removeFromVector`, die einen Vektor von Integern `vec` und einen Integer `value` übergeben bekommt und aus `vec` alle Vorkommnisse von `value` entfernt! Diese Funktion soll eine lineare Laufzeitkomplexität haben. Lässt sich diese Funktion wie `loopVector` um eine Unterstützung für andere sequentielle Container (welche?) erweitern?



#### Teilaufgabe 4

Schreiben Sie die Klasse `Leaderboard`. Diese soll eine Bestenliste von `double` Werten abbilden.

- a) Schreiben Sie einen Konstruktor für die Klasse! Dieser soll einen `size_t` Parameter erhalten, der die (maximale) Länge der Bestenliste angibt.
- b) Implementieren Sie eine Methode `add`, die einen neuen Wert vom Typ `double` dem Highscore hinzufügt! Verwenden Sie zur Speicherung der Werte einen sequentiellen Container! Ist noch nicht die maximale Anzahl an Werten im Highscore vorhanden, so soll der Wert auf jeden Fall eingefügt werden. Ist die Bestenliste bereits voll, wird der neue Wert hinzugefügt, wenn er größer als der kleinste bisher in der Bestenliste vorhandene Wert ist, der dafür entfernt werden soll.
- c) Damit die Bestenliste ausgegeben werden kann, muss es möglich sein über ihn zu iterieren. Implementieren Sie dazu die Methoden `.begin` und `.end`, die Iteratoren auf den dem Leaderboard zugrundeliegenden Container zurückgeben als `const`-Methoden. Die im Leaderboard enthaltenen Werte sollen dabei entsprechend ihres Wertes in absteigender Reihenfolge ausgegeben werden.

#### Hinweise:

- Es ist ratsam die Elemente in der Methode `add` bereits sortiert im verwendeten Container abzulegen.
- Die C++ Standardbibliothek bietet bereits Funktionen zum Suchen von Positionen in sortierten Containern, die für die Implementation von `.add` hilfreich sein können.
- Standardcontainer haben die Methoden `.rbegin` und `.rend`, die Reverse-Iteratoren auf den Container zurückgeben. Je nachdem, in welcher Reihenfolge Sie die Werte im Container ablegen wollen, kann dies hilfreich sein.