

### **'Oakberg'**

Datencontainer 'range-based-for' durchlaufen und 'auto' verwenden.

- Definieren Sie ein 'int'-Feld 'a', initialisiert mit den Werten 2, 3, 5, 7, sowie einen 'vector' 'v', initialisiert mit den selben Werten.
- Durchlaufen Sie 'a' und 'v' jeweils in einer 'range-based-for'-Schleife mit 'auto' und geben Sie das jeweilige Element aus.

## 0x04 – Übungen

---

### 'Brickgate'

Erweitern Sie Übung 'Oakberg'.

- Nutzen Sie 'auto&' in der 'range-based-for'-Schleife über den 'vector' 'v' und verdoppeln Sie den Wert jedes Elements.
- Definieren einen eigenen Typ 'it\_type' als 'const\_iterator' des Vektors und geben Sie in einer Schleife über einen const\_iterator 'it' das jeweilige Element des Iterators aus. Beachten Sie: '\*it' ist der Wert des Elements und 'cbegin' und 'cend' geben einen const\_iterator zurück.

## 0x04 – Übungen

---

### 'Hicks Bluff'

Suchen und löschen in Datencontainern mit Iteratoren.

- Suchen Sie in dem Vektor '{1,2,3,4,5}' das Element '2' und geben Sie die nächsten drei Elemente (inkl.) aus (wenn vorhanden).
- Legen Sie folgende 'unordered\_map' an  
'{ 1→'Eins', ..., 5→'Fünf' }' und suchen Sie dort nach dem Schlüssel '2'.
- Löschen Sie in obiger Map alle Elemente, deren Schlüssel größer ist als '2'.

Erweiterung:

- Nutzen Sie 'auto' so oft wie möglich.

## 0x04 – Übungen

---

### 'Ashfield'

Mapping von ISBN-Nummern zu Büchern.

- Implementieren Sie ein 'struct' 'buch', welches (vereinfacht) einen Autor und einen Titel enthält.
- Definieren Sie einen eigenen Typ 'katalog\_t' durch eine 'unordered\_map', die einen 'string' (ISBN) auf ein 'buch' abbildet. Legen Sie eine Variable 'katalog' dieses Typs an.
- Füllen Sie 'katalog' mit drei echten Büchern (und ISBN) Ihrer Wahl, z.B. '44245381X' → {'Walter Moers', 'Die 13 1/2 Leben des Käpt'n Blaubär'}, ...
- Suchen Sie in einer Schleife mit 'const auto&' alle Bücher, deren Titel länger als 30 Zeichen ist und geben Sie diese aus.

## 0x04 – Übungen

---

### **'Sparrow Town'**

Klasse zu Template erweitern.

- Erweitern Sie die Klasse 'bruch' bzw. 'fraction' aus Übung 'Ravencastle' zu einem Template.
- Testen Sie Ihre generische Klasse mit unterschiedlichen Datentypen.

## 0x04 – Übungen

---

### **'Bakeropolis'**

Klasse zu Template erweitern.

- Erweitern Sie die Klasse 'punkt' aus Übung 'Stone Ridge' zu einem Template.
- Testen Sie Ihre generische Klasse mit unterschiedlichen Datentypen.

### Hausübung

Sehen Sie sich die Member der STL-Templates 'vector', 'set', 'array' und 'unordered\_map' an, hier insbesondere die Funktionen zum Einfügen ('insert'), zum Löschen ('erase', 'clear') und zum Ersetzen ('emplace').

### Selbstkontrolle

- Ich habe alle Codes und Übungsthemen verstanden.
- Ich kann eine Klasse zu einer generischen Klasse, einem Template, erweitern (wenn es sinnvoll ist!).
- Ich kann Elemente aus Datencontainern lesen, suchen und verändern.
- Ich kann Iteratoren verwenden.