

Clicker

Bitte den Link

<https://vc2.sonia.de/b/har-2cy-qhv-bj0>

für WS 2025/26 nutzen.

Clicker für HHe

Einloggen vom Handy auf <https://vc2.sonia.de/b/har-2cy-qhv-bj0>
für WS 2025/26.

Konferenz dort ohne Audio starten.

Auch von Laptop mit anderen Namen dort einloggen und testen, ob
man antworten kann

Nun Umfrage erstellen, indem auf Plus geklickt wird

Termine des Semesters

Termin	Vorlesung	Übungen und Feedback
	Freitag ; Block 1+2	
28. Nov	STL, Iteratoren;	Bis 27.11 Teamaufgabe; erste Demo; Demo per BBB vorher
05. Dez	lineare und assoziative Container; STL Algorithmen; Algorithmus oder Methode	
12. Dez	Klasse unique_ptr, shared_ptr, Lambda-Ausdrücke; Verschiebeoperatoren ,	Bis 11.12 Teamaufgabe; Demo per BBB; Test auf unbekanntem Daten
19. Dez	Rest; Vorbereitung Klausur	Finale Abgabe Teamaufgabe Mi. 17.12; Demo per BBB vorher

Klausur: **Mi, 14.1.26**; 11:00 – 12:30 Uhr; R. 252
Klausureinsicht: Jan 2026 **xxx** Uhr

Rückblick, 12.12.2024

- hr Stand bei der Wort-Korrektur
- Verschiebeoperatoren (operator=(X&& org);)
- Intelligente Zeiger (unique_ptr)
- |
- Intelligente Zeiger (shared_ptr)

Clicker: Raten Sie mal bzw. Wissen ist besser

```
list<int> cont;  
for (int i=1; i<6; ++i) { cont.push_back(i); }  
// Ausgabe wäre nun 1 2 3 4 5  
//copy(cont.begin(), cont.end(), ostream_iterator<int>(cout, " "));  
  
// 3 mit remove entfernen  
remove(cont.begin(), cont.end(), 3);  
copy(cont.begin(), cont.end(), ostream_iterator<int>(cout, " "));
```

Ausgabe?

1. 1 2 4 5 5
2. 1 2 4 5
3. 1 2 3 4 5
4. 5 4 2 1

Clicker

```
list<int> cont;  
for (int i=1; i< 6; ++i) {  
    cont.push_back( i );  
}
```

```
// Ausgabe ist 1 2 3 4 5  
copy(cont.begin(), cont.end(), ostream_iterator<int>(cout, " "));  
remove(cont.begin(), cont.end(), 3 );  
cont.erase(cont.begin(), cont.end());  
copy(cont.begin(), cont.end(), ostream_iterator<int>(cout, " "));
```

2. Ausgabe?
1. 1 2 3 4 5
2.
3. 1 2 4 5
4. 1 2 4 5 3

Clicker

```
list<int> cont;  
for (int i=1; i< 6; ++i) {  
    cont.push_back( i );  
}
```

2. Ausgabe?

1. 1 2 3 4 5
2. 1 2 4 5 5
3. 1 2 4 5
4. 1 2 4 5 3

// Ausgabe ist 1 2 3 4 5

```
copy(cont.begin(), cont.end(), ostream_iterator<int>(cout, " "));  
cont.erase(remove(cont.begin(), cont.end(), 3), cont.end() );  
copy(cont.begin(), cont.end(), ostream_iterator<int>(cout, " "));
```

Planung heute

- Stand: Verbesserung der automatischen Transkription (Wortkorrekturaufgabe (letzter Abgabezeitpunkt 23.12.25; 23:59 Uhr)
- Lambda-Ausdrücke zumindest andeuten
- Infos zur Klausur
- Clicker zur Klausur