

Clicker

Bitte den Link

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

Termine des Semesters

| Termin | | Vorlesung | Übungen und Feedback |
|--------|---------|---|---|
| Woche | | Freitag ; Block 1+2 | |
| 5 | 24. Okt | Werte-, Zeigerparameter etc., Klassen string, vector<T> ADT; Klassen, tiefe und flache Kopie | |
| 6 | 31. Okt | fällt aus: Feiertag | Finale Abgabe Fr. 31.10 Übung 2; Demo per BBB vorher |
| 7 | 07. Nov | tiefe und flache Kopie: Kopierkonstruktor, LogTrace ; Operatoren | 7.11; Teambildung abgeschlossen |
| 8 | 14. Nov | Operatoren (Teil 2); Templates, Polymorphie | |
| 9 | 21. Nov | STL, Iteratoren; | |
| 10 | 28. Nov | lineare und assoziative Container; STL Algorithmen | Bis 27.11 Erste Teamaufgabe abgegeben; Demo per BBB vorher |
| 11 | 05. Dez | Algorithmus oder Methode Klasse unique_ptr,, shared_ptr, Lambda-Ausdrücke | |
| 12 | 12. Dez | Verschiebeoperatoren , | |
| 13 | 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

Nächste Übungsaufgabe zur Abgabe vorstellen

Aufgabe Mit30_40_Belegen (60 Minuten als Einzelübung)

Zeiger auf Zeiger `int** ptr`

Zeiger als Referenzen `f(int* & apr_zeiger) { }`

Andere nützliche Dinge über Zeiger

`const_cast`, `reinterpret_cast`

`const int* const ptr`; etc.

Clicker-“Abstimmung“

```
int j = 61; /* 1 */  
const int ka = 36; /* 2 */  
if (ka = 144) /* 3 */  
    const double PI = 2.0 * j; /* 4 */
```

Welche Codezeile ist SYNTAKTISCH falsch?

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

Clicker-“Abstimmung“

Was wird in „datei“ ausgegeben?

1. j= 42
2. j= 122
3. 42
4. 122

```
void tester1 (int& p1) {  
    p1 = 42;  
}  
  
void aufgFunk1() {  
    int j = 122;  
    tester1 (j);  
    datei << "j= " << j;  
}
```

Clicker-“Abstimmung“

Was wird in „datei“ ausgegeben?

1. [2] 11
2. [2] 2
3. [2] 3
4. [2] 126

```
void tester3 (int fld []) {  
    fld [2] = 126;  
}  
  
void aufgFunk3() {  
    int arr [] = { 11, 2, 3 };  
    tester3 (arr);  
    datei << "[2] " << arr[2];  
}
```

Clicker-“Abstimmung“

```
void mapTest() {  
    map<string, double> gewicht;  
    gewicht["Torr"] = 16.39;  
    gewicht["Torr"] = 81.60;  
    datei << gewicht["Torr"]  
        << " " << gewicht["Ida"];  
}
```

Was wird in „datei“ ausgegeben (Blanks nicht beachten)?

1. Absturz
2. 81.60 Absturz
3. 81.6 0
4. 81.60 16.39

Clicker-“Abstimmung“

```
void zeigerTest () {  
    int anzahl = 144;  
    int zahl[] = { 21, 31, 41, 17, 22, 19 };  
    int* p = &(zahl[2]);  
    datei << *p << ", ";  
    (*p)++;  
    datei << *p << ", ";  
    ++p;  
    datei << *p << ", ";  
}
```

| | | |
|------|---------|-----|
| 996 | anzahl | 144 |
| 1000 | zahl[0] | 21 |
| 1004 | zahl[1] | 31 |
| 1008 | zahl[2] | 41 |
| 1012 | zahl[3] | 17 |
| 1020 | zahl[4] | 22 |
| 1024 | zahl[5] | 19 |
| 1028 | p | ??? |

Was steht in den Adressen ab 1028 nach Ausführung des Blauen?

1. 1008
2. 1012,
3. 41,
4. 17

Clicker-“Abstimmung“

```
void zeigerTest () {  
    int anzahl = 144;  
    int zahl[] = { 21, 31, 41, 17, 22, 19 };  
    int* p = &(zahl[2]);  
    datei << *p << ", ";  
    (*p)++;  
    datei << *p << ", ";  
    ++p;  
    datei << *p << ", ";  
}
```

| | | |
|------|---------|------|
| 996 | anzahl | 144 |
| 1000 | zahl[0] | 21 |
| 1004 | zahl[1] | 31 |
| 1008 | zahl[2] | 41 |
| 1012 | zahl[3] | 17 |
| 1020 | zahl[4] | 22 |
| 1024 | zahl[5] | 19 |
| 1028 | p | 1008 |

Was steht in den Adressen ab 1008-1011 und 1028-1031 nach Ausführung des Blauen?

1. 41 / 1008
2. 42 / 1008
3. 41 / 1009
4. 41 / 1012

Clicker-“Abstimmung“

```
void zeigerTest () {  
    int anzahl = 144;  
    int zahl[] = { 21, 31, 41, 17, 22, 19 };  
    int* p = &(zahl[2]);  
    datei << *p << ", ";  
    (*p)++;  
    datei << *p << ", ";  
    ++p;  
    datei << *p << ", ";  
}
```

| | | |
|------|---------|------|
| 996 | anzahl | 144 |
| 1000 | zahl[0] | 21 |
| 1004 | zahl[1] | 31 |
| 1008 | zahl[2] | 42 |
| 1012 | zahl[3] | 17 |
| 1020 | zahl[4] | 22 |
| 1024 | zahl[5] | 19 |
| 1028 | p | 1008 |

Was steht in den Adressen ab 1008-1011 und 1028-1031 nach Ausführung des Blauen?

1. 43 / 1008
2. 42 / 1008
3. 42 / 1009
4. 42 / 1012

Clicker-“Abstimmung“

```
void zeigerTest () {  
    int anzahl = 144;  
    int zahl [] = { 21, 31, 41, 17, 22, 19 };  
    int* p = &(zahl[2]);  
    datei << *p << ", ";  
    (*p)++;  
    datei << *p << ", ";  
    ++p;  
    datei << *p << ", ";  
}
```

Was wird in „datei“ ausgegeben?

1. 31, 41, 17,
2. 41, 42, 17,
3. 41, 17, 18,
4. 41, 17, 22,

Clicker-“Abstimmung“

Welche Aussage trifft zu?

1. Header-Dateien haben üblicherweise die Endung „.h“?
2. Quellcode-Dateien haben üblicherweise die Endung „.h“?

Clicker-“Abstimmung“

Welche Aussage trifft zu?

1. Funktionen werden in einer Header-Datei deklariert (Schnittstelle angegeben)
2. Funktionen werden in der Quellcode-Datei deklariert

Clicker-“Abstimmung“

Welche Aussage trifft zu?

1. Größere Funktionen werden in der Header-Datei **implementiert /definiert**.
2. Größere Funktionen werden in der Quellcode-Datei implementiert /definiert.

Clicker-“Abstimmung“

```
#ifndef VORGANG_HEADER
#define VORGANG_HEADER

struct Vorgang {
    double dauer;
    double fruehanf;
    double spaetend;
};
void initVorgang(Vorgang* v);
void ausgabeVorgang(const Vorgang* v);
bool istDurchfuehrbar(Vorgang *v);
void passeFruehanfAn(
    Vorgang* v, const Vorgang* vv);
void passeSpaetEndAn(
    Vorgang* v, const Vorgang* vn);

#endif /* VORGANG_HEADER */
```

Vorgang.h

Wie nennt man die roten Codeteile?

1. Dateikopf
2. Include-Wächter
3. Datei-Deklaration
4. Funktionsdefinition

Clicker-“Abstimmung“

Welche Aufgabe hat ein Include-Wächter?

1. Kurzbeschreibung des Datei-Inhalts
2. Mehrfaches Einbinden / Includen der gleichen Datei in der gleichen Übersetzungseinheit (Source-Datei *.cxx) verhindern
3. Erleichterung der Funktionstests
4. Hinweise auf externe Symbole für den Linker

Prioritäten

```
void stringTest()
{
    string ch = "kkk";
    string ch1 = "geht so";
    cout << ch == ch1;
}
```

Prioritäten von Operatoren

```
void stringTest()
{
    string ch = "kkk";
    string ch1 = "geht so";
    // cout << ch == ch1; // << has higher Priority than ==
    cout << (ch == ch1); // brackets necessary
}
```

Vorlesungsplanung

Wiederholung, insbesondere Aufgaben aus dem einem Test für vorherige Semester

Klasse string

Klasse vector<T>, Klasse array<T,size>

Referenz oder Zeiger oder Werte oder doch was anderes?

Klassen beginnen

- Am Beispiel von Vektor, Konstruktor, Destruktor
- Konstruktoren
- +Z, -Z
- LogTrace / FunctionLog