

# Objektorientierte Programmierung mit C++:

## Wiederholung

Profs. M. Dausmann, D. Schoop, A. Rößler

Fakultät Informationstechnik, Hochschule Esslingen



## Was ist ein Strommanipulator?

Ein Strommanipulator liegt zwischen den „<<“ Operanden und dem Ausgabe- oder Eingabestrom und mit dessen Hilfe kann der Strom formatiert werden

```
cout << Strommanipulator << "Text" << std::endl;
```



# Welche Streams gibt es in C++?

```
cin >> Eingabevariable;    // Standardeingabe  
cout << Ausgabevariable;   // Standardausgabe  
cerr << Ausgabevariable;   // ungepufferte Fehlerausgabe  
clog << Ausgabevariable;   // gepufferte Fehlerausgabe
```



# Was ist ein Streamstatus?

Jeder Stream hat einen Status für die Anzeige von Fehlern und außergewöhnlichen Zuständen.

```
eof() // gibt true bei end-of-file (ios::eofbit gesetzt)
fail() // gibt true bei Fehler (ios::failbit oder ios::badbit)
bad() // gibt true bei fatalem Fehler (ios::badbit gesetzt)
good() // gibt true, wenn Stream OK ist (ios::goodbit)
```

# Frage 3



<u>iostate</u> value (member constant)	indicates	functions to check state flags				
		<u>good()</u>	<u>eof()</u>	<u>fail()</u>	<u>bad()</u>	<u>rdstate()</u>
goodbit	No errors (zero value <u>iostate</u> )	true	false	false	false	goodbit
eofbit	End-of-File reached on input operation	false	true	false	false	eofbit
failbit	Logical error on i/o operation	false	false	true	false	failbit
badbit	Read/writing error on i/o operation	false	false	true	true	badbit

## Frage 4

---



# Welche Bibliotheken sind für das Lesen und Schreiben von Dateien einzubinden?

- `ifstream` zum Lesen aus einer Datei
- `ofstream` zum Schreiben in eine Datei
- `fstream` zum Lesen und Schreiben

## Frage 5

---



# Welche Modis können beim Schreiben und Lesen von Dateien gewählt werden?

<b>in</b>	input	Lesen ab Dateianfang
<b>ate</b>	at end	Positionieren auf Dateiende
<b>binary</b>	binary	Bearbeiten ohne Aufbereitung
<b>out</b>	output	Schreiben ab Dateianfang
<b>app</b>	append	Schreiben hinter Dateiende
<b>trunc</b>	truncate	Überschreiben des alten Inhalts



# Was ist eine Binärdatei?

- keine Textdatei ist, die also nicht nur Repräsentationen ‚druckbarer‘ Zeichen enthält, sondern die beliebige Bitmuster enthalten kann.
- Es gibt auch die Möglichkeit, die binäre Darstellung aus dem Speicher direkt in eine Datei zu schreiben und umgekehrt. Man spricht dann von **Binärdateien**.
- Beispiel: Audio- oder Bild-Dateien





# Was ist ein Namensraum?

Ein Namensraum definiert eine Scope in der sich Programmbausteine (Methoden, Konstanten, Klassen) befinden und nach außen nicht sichtbar ist.

Diese sind nur innerhalb des Namensraumes gültig



# Was ist `size_t` für ein Zahlentyp?

Ein Typ für nicht negative ( ab 0 ) ganze Zahlen wie  
*unsigned int*

*Vorteil:*

*size\_t* ist immer groß genug, um die Größe des  
größtmöglichen primitiven Datentyps aufzunehmen

## Frage 11

---



**Wie kann ich bei call by reference  
eine implizite Typenumwandlung  
erreichen?**

**Das geht in C++ nicht!**



# Was ist der Unterschied zwischen Call by Value und Call by Reference?

**Call by value:** Hier wird der Parameter einer Methode kopiert und übergeben

**Call by reference:** Hier wird die Referenz eines Parameters übergeben (Vorsicht: Es werden Daten ausserhalb der Methode manipuliert)



# Was bedeutet Overloading bei Methoden und welche Regeln sind zu beachten?

Es können Methoden mit gleichem Namen, aber unterschiedlicher Parameterliste definiert werden

- Die **Signatur** der Funktion muss verschieden sein, d.h. gleicher Name aber verschiedene Parameterliste.
- Der Rückgabetyt spielt keine Rolle. D. h. es gibt keine Überladung mit gleicher Signatur und verschiedenen Rückgabetypen.



# Was ist der Unterschied zwischen einem **Pointer** und einer **Referenz**?

Eine **Referenz** repräsentiert eine Adresse im Speicher.

Ein **Pointer** zeigt auf eine Adresse (Pointer-Wert) im Speicher.