

Prof. Dr.-Ing. Hartmut Helmke  
in  
Deutsches Zentrum für  
Luft- und Raumfahrt e.V. (DLR)  
Institut für Flugführung  
Abteilung Lotsenassistenzsysteme  
Postfach 32 67  
38108 Braunschweig

Tel. 0531 **295-2599**

E-Mail: **Hartmut.Helmke@DLR.DE**

# Vorstellung der eigenen Person

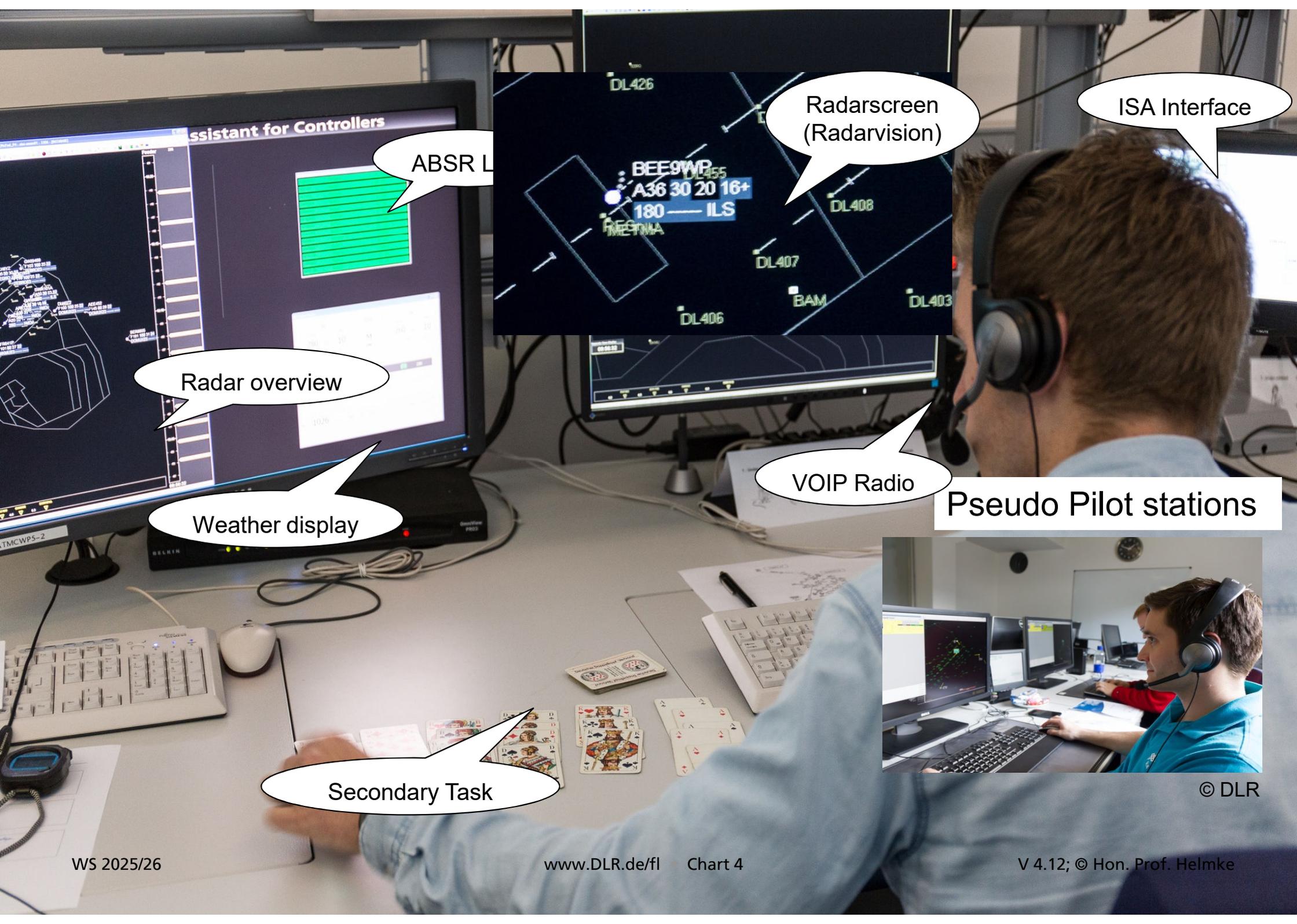
—

Flugzeuge über Europa etc.

"Q:\AndereDinge\Slide1\_OneDay\_Anim.avi" mit VLC Player öffnen

"Q:\AndereDinge\12StundenIn2Minuten.wmv" mit VLC oder Windows  
Media Player

"Q:\AndereDinge\FlugverkehrUSA2008.pptx" in Power Point abspielen



Assistant for Controllers

ABSRL

Radar overview

Weather display

Secondary Task

Radarscreen (Radarvision)

ISA Interface

VOIP Radio

Pseudo Pilot stations

© DLR

## Summary

Results of the Dusseldorf trials<sup>\*\*\*</sup> are with respect to:

### Airlines:

- 50 to 65 liters savings of kerosene per flight in medium traffic situations\*

### - Airports:

- Increase of flow by 1 to 2 landings per hour

### - ANSPs:

- Reduction of controller workload needed for advisory input by a factor of 2 to 3
- Reduced head down times increase safety

### - Relevance for society:

- Saving of about 130 kg of CO<sub>2</sub> per flight\*\*

\* A320,

\*\* 0.8 kg / l, 1 kg kerosene results in 3.15 kg CO<sub>2</sub>; 35 landings per hour

\*\*\* extrapolation of results of 60 minutes scenarios for 23R,  
8 controllers..., see previous slides

# Termine des Semesters

		Termin	Vorlesung	Übungen und Feedback
Vorles Woche		Freitag ; Block 1+2		
1	1	26. Sep	Einführung, Worthäufigkeiten in einer Datei zählen: Datentyp., Steueranw.	
	2	03. Okt	fällt aus: Feiertag	
2	3	10. Okt	Funktionen Strukturen; Test first; Header-Dateien, Include-Wächter, Namespaces, Werte- und Ref-Semantik, Zeiger, Heap- und Stackspeicher	Abgabe Fr 17.10; Erste einfache Übung 5% Pkte
3	4	17. Okt	Werte-, Zeigerparameter etc., Klassen string, vector<T>	
4	5	24. Okt		
	6	31. Okt	fällt aus: Feiertag	Finale Abgabe Fr. 31.10 Übung 2; Demo per BBB vorher
5	7	07. Nov	tiefe und flache Kopie: Kopierkonstruktor, LogTrace ; Operatoren	7.11; Teambildung abgeschlossen
6	8	14. Nov	Operatoren (Teil 2); Templates, Polymorphie	
7	9	21. Nov	STL, Iteratoren;	
8	10	28. Nov	lineare und assoziative Container; STL Algorithmen Algorithmus oder Methode Klasse unique_ptr,, shared_ptr, Lambda-Ausdrücke	Bis 27.11 Erste Teamaufgabe abgegeben; Demo per BBB vorher
9	11	05. Dez		
10	12	12. Dez	Verschiebeoperatoren ,	
11	13	19. Dez	Rest; Vorbereitung Klausur	Finale Abgabe Teamaufgabe Mi. 17.12; Demo per BBB vorher

## **BBB**

Link zu BBB für WS 2025/26:

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

## Termine des Semesters (2)

- Die Vorlesungsinhalte sind ein (erster) Plan
- Die Termine sind mehr oder weniger fix (ich wäre flexibel)
- Wir haben 13 Wochen, aber nur 11 Vorlesungen

	Termin	Vorlesung	Übungen und Feedback
		Freitag ; Block 1+2	
Vorles Woche			
1	1	26. Sep Einführung, Worthäufigkeiten in einer Datei zählen.	
	2	03. Okt Datentyp, Steueranw.	
		<b>fällt aus: Feiertag</b>	
2	3	10. Okt Funktionen Strukturen, Test first, Header-Dateien,	
		Include-Wächter, Namespaces, Werte- und Ref-Semantik, Zeiger, Heap- und	
3	4	17. Okt Stack Speicher	Abgabe Fr 17.10; Erste einfache Übung 5% Pkte
		Werte-, Zeigerparameter etc., Klassen string,	
4	5	24. Okt vector<T>	
	6	31. Okt <b>fällt aus: Feiertag</b>	Finale Abgabe Fr. 31.10 Übung 2; Demo per BBB vorher
		tiefe und flache Kopie: Kopierkonstruktor, LogTrace ;	
5	7	07. Nov Operatoren	7.11; Teambildung abgeschlossen
6	8	14. Nov Operatoren (Teil 2); Templates, Polymorphie	
7	9	21. Nov STL, Iteratoren;	
			Bis 27.11 Erste Teamaufgabe abgegeben; Demo per BBB vorher
8	10	28. Nov lineare und assoziative Container; STL Algorithmen	
		Algorithmus oder Methode Klasse unique_ptr, shared_ptr, Lambda-Ausdrücke	
9	11	05. Dez	
		12. Dez Verschiebeoperatoren ,	
10	12		
		19. Dez Rest, Vorbereitung Klausur	Finale Abgabe Teamaufgabe Mi. 17.12; Demo per BBB vorher
11	13		

- In der Vorlesung ist der Stoff damit etwas „Komprimierter“.
- Es gibt Übungen und bewertete Abgaben.
- Abnahme, versuche ich per BBB mit jedem „einzeln“ durchzuführen
- Zweiergruppen sind zu Beginn möglich. Ich möchte bei den BBB-Abgaben aber sehen, wer was gemacht hat.
- Ziel ist, Sie auch an die „Programmierung im Großen“ heranzuführen (im Rahmen der 5 Credits)

## Termine des Semesters (3)

- Ziel ist, Sie auch an die „Programmierung im Großen“ heranzuführen
- Der nach der zweiten Abgabe Zusammenfassung in Teams à 4 bis 6 Mitglieder
- Ist aufwändiger als in 2er-Teams, daher mehr Punkte

	Termin	Vorlesung	Übungen und Feedback
Vorles Woche		Freitag : Block 1+2	
1	1 26. Sep	Einführung, Worthäufigkeiten in einer Datei zählen.	
	2 03. Okt	Datentyp., Steueranw. fällt aus: Feiertag	
2	3 10. Okt	Funktionen Strukturen; Test first, Header-Dateien, Include-Wächter, Namespaces,	
3	4 17. Okt	Werte- und Ref-Semantik, Zeiger, Heap- und Stackspeicher	Abgabe Fr 17.10; Erste einfache Übung 5% Pkte
4	5 24. Okt	Werte-, Zeigerparameter etc., Klassen string, vector<T>	
	6 31. Okt	fällt aus: Feiertag	Finale Abgabe Fr. 31.10 Übung 2; Demo per BBB vorher
5	7 07. Nov	tiefe und flache Kopie: Kopierkonstruktor, LogTrace, Operatoren	7.11; Teambildung abgeschlossen
6	8 14. Nov	Operatoren (Teil 2), Templates, Polymorphie	
7	9 21. Nov	STL, Iteratoren;	
8	10 28. Nov	lineare und assoziative Container; STL Algorithmen Algorithmus oder Methode Klasse unique_ptr,, shared_ptr, Lambda-Ausdrücke	Bis 27.11 Erste Teamaufgabe abgegeben; Demo per BBB vorher
9	11 05. Dez		
10	12 12. Dez	Verschiebeoperatoren ,	
11	13 19. Dez	Rest, Vorbereitung Klausur	Finale Abgabe Teamaufgabe Mi. 17.12; Demo per BBB vorher

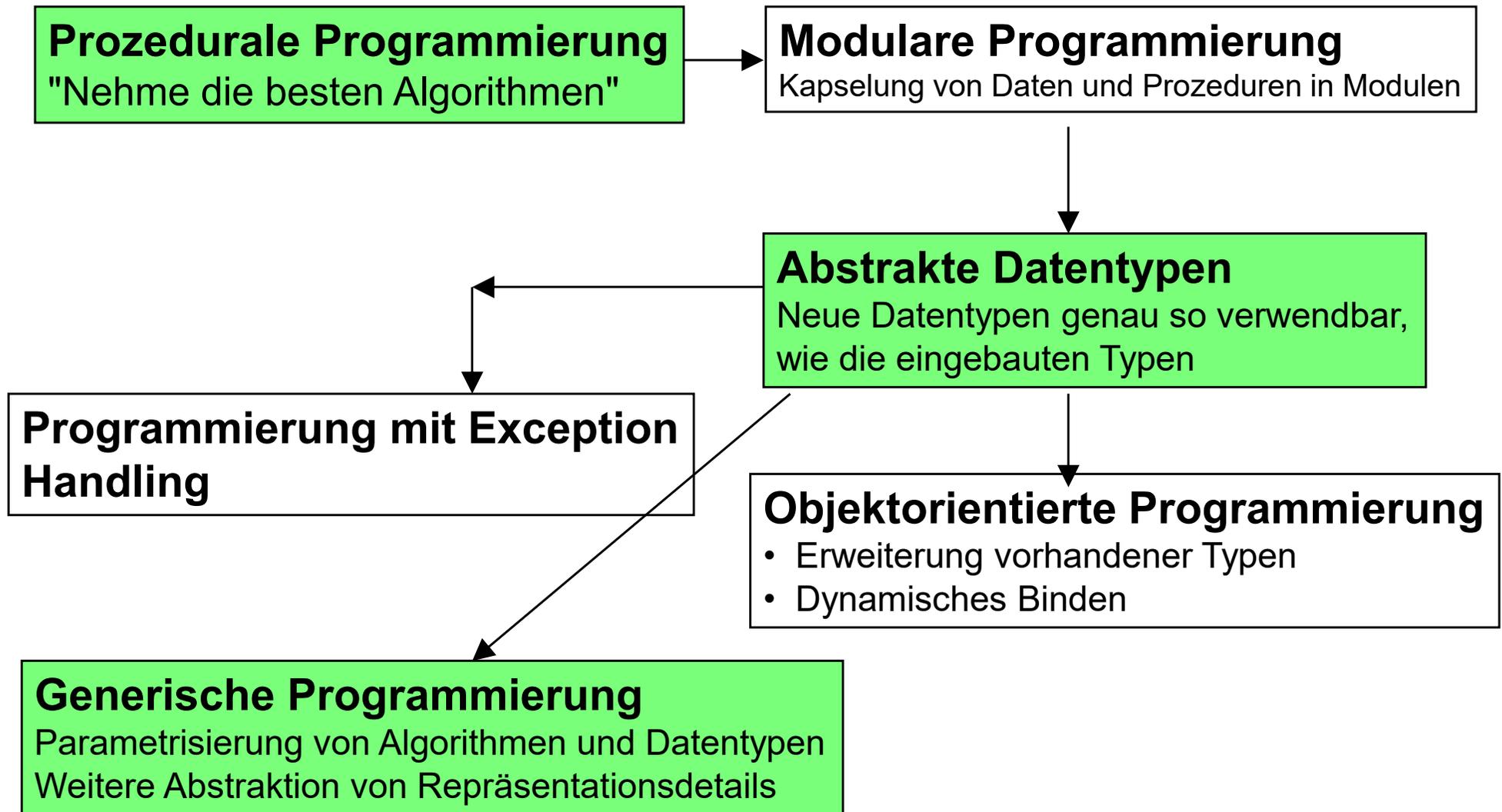
- Zur Abgabe nutze ich SVN; hat für mich Vorteile gegenüber GIT; Intern werden Sie vermutlich GIT nutzen
- Nächste Vorlesung am 10.10
- Erste Abgabe am 17.10 (gerne auch früher per BBB-Rückmeldung)
- Ggf. habe ich noch einen Hiwi zur Unterstützung

## **E-Mail an mich: [Hartmut.Helmke@dlr.de](mailto:Hartmut.Helmke@dlr.de)**

Bitte E-Mail an mich, wenn Sie an der Vorlesung teilnehmen wollen.  
x Namen habe ich schon. Wenn ich Ihre Daten schon habe, habe ich Ihnen  
eine E-Mail geschrieben.

Für SVN-Zugriff-Beantragung brauche ich den Benutzernamen  
(idXXXXXX) und die Matrikelnummer.

# Programmierparadigmen von C++



## Übungen: „Rufzeichenextraktion aus Funksprüchen“

good morning **air france one two alfa** descend flight level eight zero  
→ AFR12A

Was tun, wenn nun aber nur AFR122A in der Luft ist? Wir suchen das ähnlichste

Ähnlichkeit von Wortfolgen:

- „air france one two alfa“
- „air france one two two alfa“
- Levenshtein-Distanz von 1

Das war im letzten Semester  
(und davor, und auch davor).

Hauptsächlich wird es um die Implementierung von Datenstrukturen mit Hilfe der STL zum effizienten Suchen gehen.

Wer findet mehr als 90% Treffer in echten Funksprüchen

Allein, Zwei-Teams oder in Gruppen bis 6 (Jeweils andere Herausforderungen).

## Übungen: „Raten“ von Wegpunktnamen

Gesagt: highlight **rapor**

Erkannt: height light **rapo**

Oder: highlight **papa**

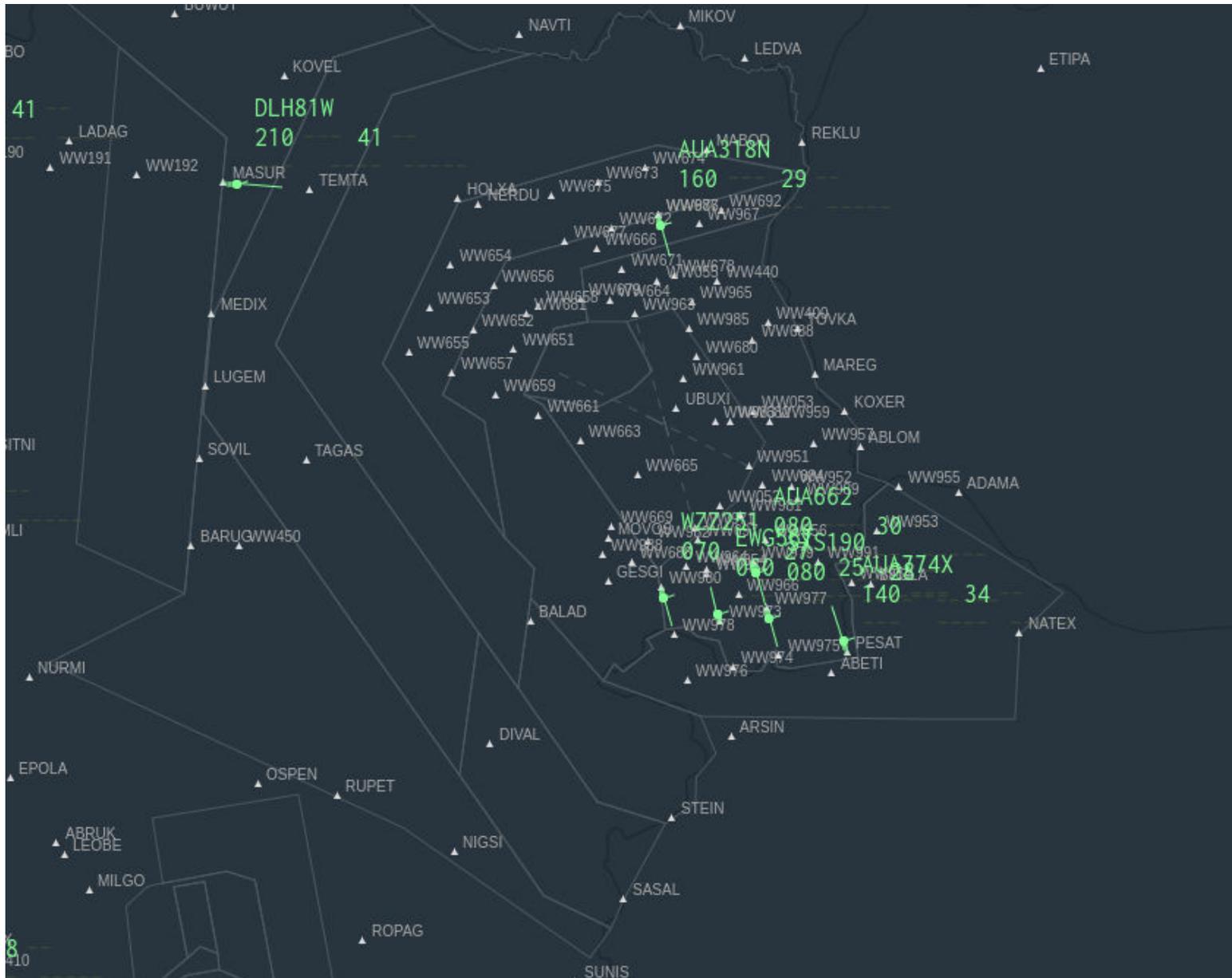
Oder highlight **rapot**

Die Namen der möglichen Wegpunkte sind bekannt.

Wegpunkt sind in der Regel Kunstworte, die aus (meist) 5 Buchstaben bestehen, wovon 2-3 Vokale sind und 3 bis 2 Konsonanten.

Die Levenshtein-Distanz wird hier wieder eine Rolle spielen, aber dieses Mal auf Buchstaben und nicht auf Wortebene

# Die verschiedenen Programmierparadigmen von C++



LUGEM  
MEDIX  
BALAD  
STEIN  
SASAL

...

## Levenshtein-Distanz auf Buchstaben-Ebene

Gesagt:    h i g h    l i g h t r a p o r

Erkannt:   h e i g h t   l i g h t r a p o

→        Levenshtein Distanz von 3 bzw. für den Wegpunkt von 1.

## Know-how-Abfrage

Bei den grünen Abfragen wird erwartet, dass Sie sich das Wissen unabhängig von dieser Vorlesung aneignen bzw. es schon aus den vorhergehenden Semestern besitzen.

Kennen Sie Arrays?

1. Ja
2. Nein
3. so ungefähr

Link zu BBB:

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

## Clicker-“Abstimmung“

Kennen Sie die Aufgaben von Compiler und Linker?

1. Ja
2. Nein
3. so ungefähr

Kennen Sie den Unterschied zwischen einer Gleitkommazahl (Typ REAL bzw. float, double) und einer ganzen Zahl (Typ Integer)?

1. Ja
2. Nein
3. so ungefähr

Kennen Sie Arrays?

1. Ja
2. Nein
3. so ungefähr

## Clicker-“Abstimmung“ (2)

Kennen Sie Records bzw. Strukturen?

1. Ja
2. Nein
3. so ungefähr

Kennen Sie Aufzählungstypen (Enumerations)?

1. Ja
2. Nein
3. so ungefähr

## Clicker-“Abstimmung“ (3)

Könnten Sie jetzt in Java eine Funktion, die das minimale Element in einem Array ermittelt, implementieren?

Wie lange bräuchten Sie dafür (Schätzung)?

\_\_\_ <5 Minuten

\_\_\_ <10 Minuten

\_\_\_ < 20 Minuten

> 20 Minuten

ich würde mir zutrauen, die Lösung jetzt an der Tafel vorzuführen

Ja

Nein

Mit Hilfe würde es schon klappen

## Clicker-“Abstimmung“ (4)

Könnten Sie jetzt in Java eine Funktion implementieren, die ermittelt, ob ein String ein Palindrom ist (Palindrom: Zeichenkette, die von vorn und von hinten gelesen gleich bleibt, z.B. ABBA, hammah, xbkbx)?

Wie lange bräuchten Sie dafür (Schätzung) ?

\_\_\_ < 10 Minuten

\_\_\_ <15 Minuten

\_\_\_ 25 Minuten

> 25 Minuten

ich würde mir zutrauen, die Lösung jetzt an der Tafel vorzuführen

Ja

Nein

Mit Hilfe würde es schon klappen

## Clicker-“Abstimmung“ (5)

Welche mittlere Laufzeit-Komplexität hat Headp-Sort,  
um ein Array von Zahlen zu sortieren?

(1)  $O(1)$  (2)  $O(\log N)$  (3)  $O(N)$  (4)  $O(N \log N)$  (5)  $O(N * N)$  (6)  $O(N*N*N)$

Kennen Sie sich mit der hexadezimalen Schreibweise von Zahlen aus?

1. Ja
2. Nein
3. so ungefähr

Geben Sie an, welcher dezimalen Zahl die hexadezimale  
Schreibweise „A6“ entspricht.

1. 106
2. 166
3. 172
4. 256
5. 262

## Clicker-“Abstimmung“ (6)

Werden Klassenobjekte in Java als Werte oder als Referenzen an Funktion übergeben?

(1) Als Werte  (2) Als Referenzen

Kenntnisse im Umgang mit den verschiedenen Kontrollstrukturen  
Kennen Sie die If-Then-Else-Kontrollstruktur?

(1) Ja  (2) Nein  (3) so ungefähr

Kennen Sie die for-Schleife

(1) Ja  (2) Nein  (3) so ungefähr

Kennen Sie die While-Schleife?

(1) Ja  (2) Nein  (3) so ungefähr

Kennen Sie die Repeat bzw. Do-Schleife?

(1) Ja  (2) Nein  (3) so ungefähr

## Clicker-“Abstimmung“ (7)

Wissen Sie was Zeiger in Zusammenhang mit Speicheradressen sind?

(1) Ja  (2) Nein  (3) so ungefähr

Wissen Sie was Präprozessor-Anweisungen sind?

(1) Ja  (2) Nein  (3) so ungefähr

Kennen Sie den Unterschied zwischen den beiden folgenden Anweisungen der Programmiersprache C/C++/Java?

```
zahl = i ++;
```

```
zahl = ++ i;
```

(1) Ja  (2) Nein  (3) so ungefähr

Welchen Wert hat zahl2 in dem folgenden Programmfragment am Ende?

```
int zahl = 17; int zahl2 = zahl++;
```

(1) 16 (2) 17 (3) 18 (4) 19

## Clicker-“Abstimmung“ (8)

Kennen Sie den Begriff und die Bedeutung von abstrakten Datentypen (Information Hiding)?

(1) Ja  (2) Nein  (3) so ungefähr

Wissen Sie, was Klassen bzw. Objekte sind?

(1) Ja  (2) Nein  (3) so ungefähr

Wissen Sie, was Polymorphie (Funktionsüberladen in Vererbungs-Hierarchien) ist?

(1) Ja  (2) Nein  (3) so ungefähr

Können Sie den Begriff „test first“ (oder Think, Red Bar, Green Bar, Refactor) erklären?

(1) Ja  (2) Nein  (3) so ungefähr

## Clicker-“Abstimmung“ (9)

An wie viel der voraussichtlich 4\*13 Vorlesungs- + Übungsstunden planen Sie, teilzunehmen?

\_\_\_\_\_ Stunden

(1)  $\geq 40$

(2)  $\geq 32$

(3)  $\geq 20$

(4)  $< 20$

Sind Sie ein Student/Studentin im Dualen Studium?

(1) Ja

(2) Nein

Nehmen Sie auch an der Vorlesung von Prof. Justen „Einführung in C in diesem Semester teil?

(1) Ja

(2) Nein

(3) Weiß noch nicht

## Clicker-“Abstimmung“ (10)

Erfahrung mit SVN oder GIT

(1) Ja       (2) Nein

Erfahrung mit SVN

(1) Ja       (2) Nein

## Literatur

Die Kopien (fast) aller verwendeten Folien werden vor Beginn der Vorlesung (in der Regel 2 Tage vorher) auf der „Homepage“ von Helmke zur Verfügung gestellt:

<https://www.ostfalia.de/hochschule/fakultaeten/fakultaet-informatik/fakultaetsteam/hon-prof-hartmut-helmke>

Bzw. <https://public.ostfalia.de/~helmke/>

Nach der Vorlesung gibt es ab und zu auch noch **Aktualisierungen**.

Es sind aber nur Folienkopien. Sie erleichtern das Mitschreiben. Zum Nachlesen sind sie aber weniger geeignet. Hierfür sind Lehrbücher gedacht.

Sie als Studenten der Informatik sollen hier u.a. **lernen zu lernen**.

Außer der Mathematik wird der Rest des Stoffes in 10 Jahren vermutlich Geschichte sein.

## Die verschiedenen Programmierparadigmen von C++

In der Bibliothek noch vorhanden,  
sonst vergriffen.  
Deckt C++-98 mit TN 2003 ab

Ca. 55% habe ich als Skript auf  
die Homepage gestellt.  
Die letzten Vorlesungen sind  
nicht enthalten.



## Die C-Sprachen-Großfamilie

**K&R-C**

**ANSI/ISO-C89**

**ANSI/ISO-C99**

**ISO/IEC 98899:2011 C-11)**

**ANSI/ISO-C++98 (ISO/IEC 14882:1998)**

**C++-03 (ISO/IEC 14882:2003) C++-98 klarer formuliert**

**C++ mit TR1 (April 2006)**

**ANSI/ISO-C++11 (ISO/IEC 14882:2011)**

**ANSI/ISO-C++14 (ISO/IEC 14882:2014, Jan. 2015)**

**C++17 (ISO/IEC 14882:2017, Dez 2017)**

**C++-20, C++-23, C++-26**

**Java**

**C#**

## C++-26

Status siehe z.B.: <https://isocpp.org/std/status>

Potenzielle neue große Features von C++26

- Pattern Matching,
- Reflection und
- Contracts.

## Ziele der Veranstaltung

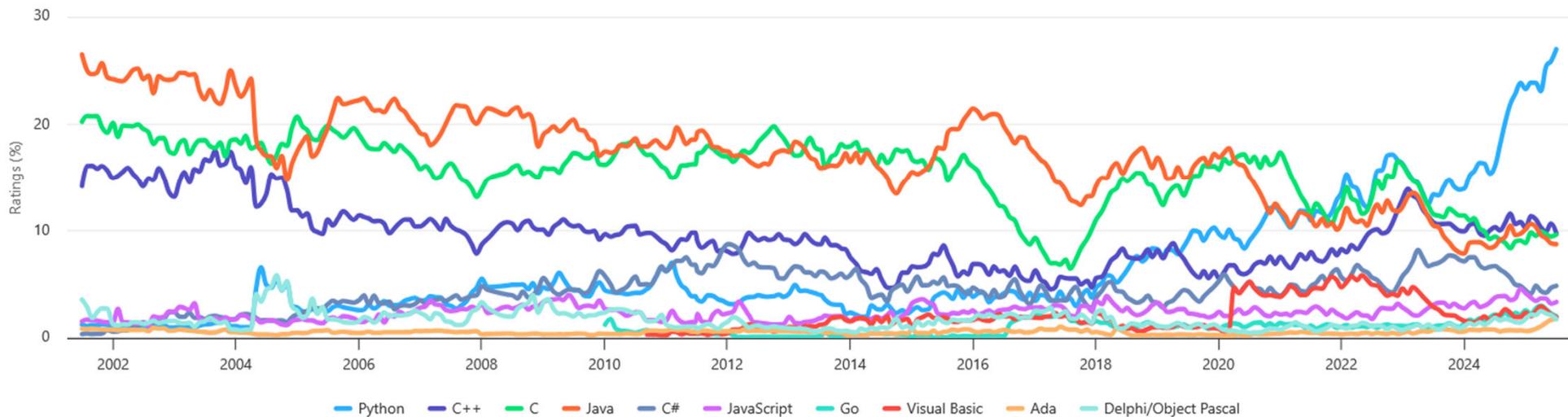
- Testgetriebene Software-Entwicklung (Test first)
- „Erst denken, dann hacken“ (zunächst (mit Worten oder formal) beschreiben)
- Die vier Gesichter von C++:
  - Prozedurale Entwicklung in C++ (Zeigermodell, Werte- und Referenzsemantik)
  - Objektorientierte Software-Entwicklung (Vererbung, Polymorphie)
  - Generische Software-Entwicklung (Templates)
  - Programmierung mit der Standard-Template-Library
- Einstieg in die Programmierung im Großen (Bibliotheken, SW-Entwicklung im Team)
- Motivation für eine Bachelorarbeit / Praxissemester im Bereich „Verstehen von *Lotsen-Sprech*“.

# TIOBE Index 26. Juli 2025

Rank	Jul 2025	Jul 2024	Change	Programming Language	Ratings	Change
1	1	1		 Python	26.98%	+10.85%
2	2	2		 C++	9.80%	-0.53%
3	3	3		 C	9.65%	+0.16%
4	4	4		 Java	8.76%	+0.17%
5	5	5		 C#	4.87%	-1.85%

TIOBE Programming Community Index

Source: www.tiobe.com



Quelle: <https://www.tiobe.com/tiobe-index/>

## Die verschiedenen Programmierparadigmen von C++

# Juli 2022 (unten) gegenüber Juli 2025 (oben rechts)

Jul 2024	Jul 2023	Change	Programming Language	Ratings	Change
1	1		 Python	16.12%	+2.70%
2	3	▲	 C++	10.34%	-0.46%
3	2	▼	 C	9.48%	-2.08%
4	4		 Java	8.59%	-1.91%

Jul 2025	Jul 2024	Change	Programming Language	Ratings	Change
1	1		 Python	26.98%	+10.85%
2	2		 C++	9.80%	-0.53%
3	3		 C	9.65%	+0.16%
4	4		 Java	8.76%	+0.17%
5	5		 C#	4.87%	-1.85%

Jul 2022	Jul 2021	Change	Programming Language	Ratings
1	3	▲	 Python	13.44%
2	1	▼	 C	13.13%
3	2	▼	 Java	11.59%
4	4		 C++	10.00%

Aug 2025	Aug 2024	Change	Programming Language	Ratings	Change
1	1		 Python	26.14%	+8.10%
2	2		 C++	9.18%	-0.86%
3	3		 C	9.03%	-0.15%
4	4		 Java	8.59%	-0.58%
5	5		 C#	5.52%	-0.87%

## Die verschiedenen Programmierparadigmen von C++

Programming Language	2025	2020	2015	2010	2005	2000	1995	1990	1985
Python	1	3	7	7	8	24	21	-	-
C++	2	4	3	4	3	2	1	2	11
Java	3	1	2	1	2	3	-	-	-
C	4	2	1	2	1	1	2	1	1
C#	5	5	5	6	10	10	-	-	-
JavaScript	6	7	8	9	11	7	-	-	-
Go	7	15	57	184	-	-	-	-	-
Visual Basic	8	14	11	-	-	-	-	-	-
SQL	9	9	-	-	-	-	-	-	-
Delphi/Object Pascal	10	182	12	10	7	-	-	-	-
Fortran	11	33	31	25	16	19	5	3	6
PHP	12	8	6	3	5	25	-	-	-
Ada	20	35	32	26	18	17	6	7	3
Lisp	27	29	27	17	15	9	7	5	2
(Visual) Basic	-	-	-	5	6	4	3	4	4

Quelle: <https://www.tiobe.com/tiobe-index/>