

Computer Vision
and Geometry Lab

Informatik I for D-MAVT

Exercise Session 1

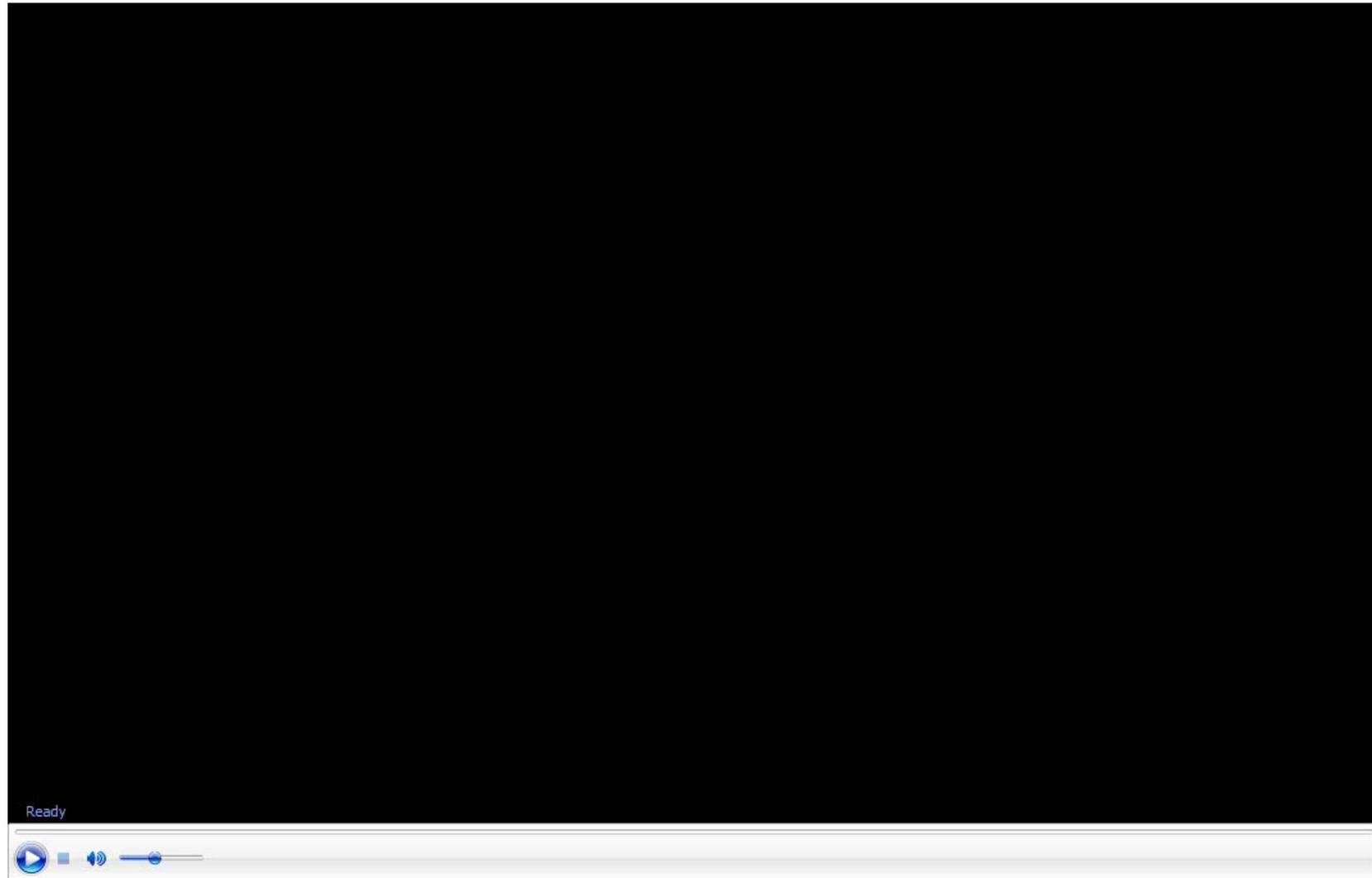
Organisatorisches

- Teaching Assistant
 - Alexander Schwing (aschwing@student.ethz.ch)
 - CAB G 89
- Übungsabgabe
 - Auf Papier, keine Emails
 - Später wenn Programme umfangreicher werden, ev. auch Abgabe per Email möglich
- Testatbedingungen
 - 75% aller Übungen
 - Voraussichtlich 12 Serien (d.h. 9 gelöste Serien)

Übungsstundenablauf

- Wünsche?
- Repetition der Vorlesungsunterlagen?
- Lediglich Vor- und Nachbesprechung der Übungen?
- Interaktiv?

Computer Crash-Kurs

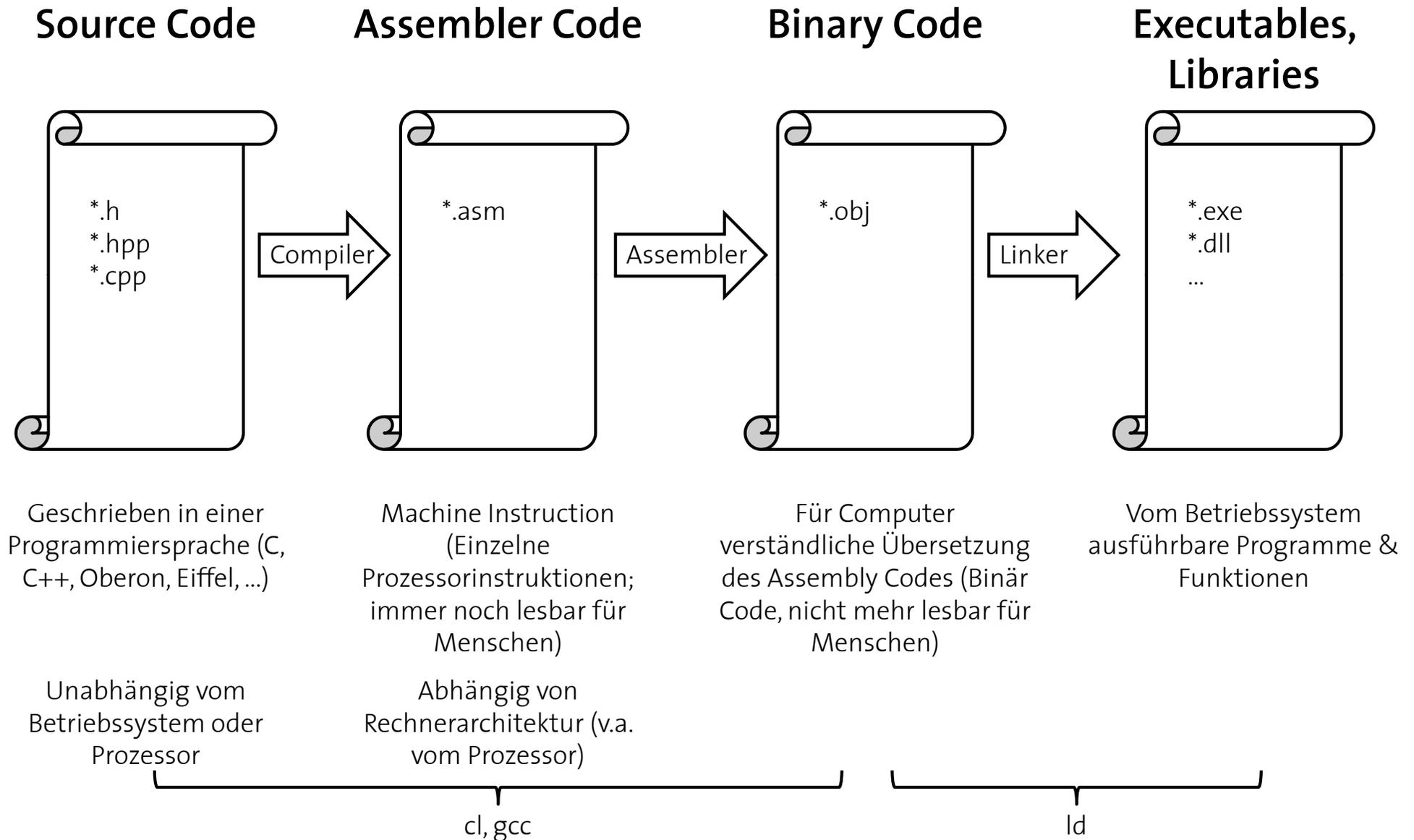


- <http://msdn.microsoft.com/en-us/beginner/bb308732.aspx>

Programmiersprachen

- Unterscheidung zwischen
 - Interpretierten Sprachen
 - Bsp: Matlab, Python, ...
 - Kompilierten Sprachen
 - Bsp: C, C++, Pascal, ...

Kompilierte Sprachen



Source Code (Bsp in C++)

■ Source Code

```
#include <iostream>
int main(int argc, char** argv)
{
    std::cout << "hello world";
    return 0;
}
```

■ Assembly Code (Auszug)

```
push    ebp
mov     ebp, esp
sub     esp, 192 ; 000000c0H
push    ebx
push    esi
push    edi
lea     edi, DWORD PTR [ebp-192]
mov     ecx, 48 ; 00000030H
mov     eax, -858993460 ; ccccccccH
rep stosd
push    OFFSET ??_C@_0M@LACCCNMM@hello?5world?$AA@
mov     eax, DWORD PTR __imp_?cout@std@@3V?$basic_ostream@DU?$char_traits@D@std@@@1@A
push    eax
call   $basic_ostream@DU
add     esp, 8
```

Source Code (Bsp in C++)

■ Binary Code vs. Assembly Code (Auszug)

00000	55	push	ebp
00001	8b ec	mov	ebp, esp
00003	81 ec co 00 00 00	sub	esp, 192; 000000coH
00009	53	push	ebx
0000a	56	push	esi
0000b	57	push	edi
0000c	8d bd 40 ff ff ff	lea	edi, DWORD PTR [ebp-192]
00012	b9 30 00 00 00	mov	ecx, 48; 00000030H
00017	b8 cc cc cc cc	mov	eax, -858993460; ccccccH
0001c	f3 ab	rep stosd	

Binary Code

Assembly Code

■ Beispiel in Visual Studio

Source Code im Detail

■ Source Code

```
#include <iostream>
int main(int argc, char** argv)
{
    std::cout << "hello world";
    return 0;
}
```

■ Include-Direktive

- Compilerdirektive: wird vom Präprozessor abgearbeitet
 - Sucht Header-Datei `iostream`
 - Kopiert Inhalt der Header-Datei anstelle des `#include` Befehls
 - Dies geschieht in einer temporären Datei, ursprünglicher Source Code wird nicht verändert
 - Temporäre Datei wird an Compiler weiter gereicht
- Wieso Header-Dateien?
 - Siehe später in der Vorlesung (oder in meiner Übungsstunde...)

Source Code im Detail

■ Source Code

```
#include <iostream>
int main(int argc, char** argv)
{
    std::cout << "hello world";
    return 0;
}
```

■ main-Funktion: Einstiegspunkt des Programs

- Betriebssystem lädt Executable in Speicher
- CPU-Programpointer wird auf Anfang der main-Funktion gesetzt
- Prozessor startet, das Executable auszuführen
- Input Argumente: Kommandozeilenparameter
 - `Int argc`
 - Anzahl der Parameter
 - Erster Parameter ist gleich dem Namen der Executable
 - `Char** argv`
 - Einzelne Parameter als Zeichenfolge (sogenanntes Array of Characters)

Source Code im Detail

■ Source Code

```
#include <iostream>
int main(int argc, char** argv)
{
    std::cout << "hello world";
    return 0;
}
```

■ Rückgabebetyp der Funktion ist ein Integer (`int`)

■ Return 0

- Gibt an, dass Programm ohne Fehler beendet wurde (Konvention, irgendein Wert könnte zurückgegeben werden...)

■ Ausgabe des Texts mittels output-stream operator `<<` des `std::cout` object

- Funktionen dieser Klasse werden in `iostream` zur Verfügung gestellt
- Darum muss diese Header-Datei eingebunden werden

Aufgabe 1

■ Source Code

```
#include <iostream>

int main(int argc, char** argv)
{
    std::cout << "Hello World" << std::endl;
    return 0;
}
```

```
#include <iostream>
using namespace std;
int main(int argc, char **argv)
{
    cout << "Hello World" << endl;
    return 0;
}
```

Aufgabe 2

■ Beispiel:

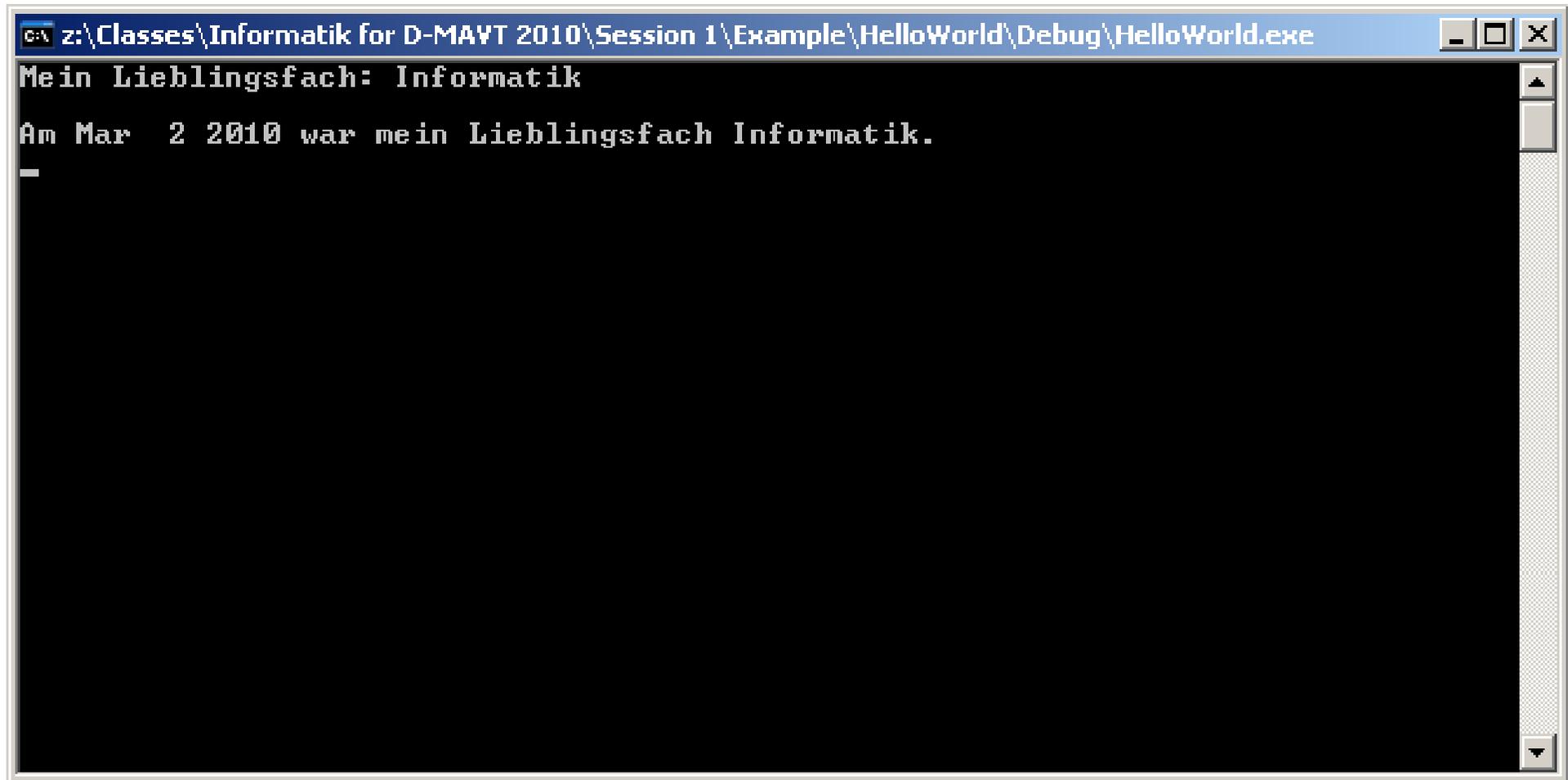
```
1  #include <iostream>
2  #include <string>
3  int main(int argc, char **argv) {
4      std::string MeinLieblingsfach;
5      std::cout << "Mein Lieblingsfach: ";
6      std::cin >> MeinLieblingsfach;
7      std::cout << std::endl;
8      std::cout << "Am " << __DATE__;
9      std::cout << " war mein Lieblingsfach " << MeinLieblingsfach << ".";
10     std::cout << std::endl;
11     return 0;
12 }
```

■ Ausgabe

Mein Lieblingsfach: Informatik

Am Mar 2 2010 war mein Lieblingsfach Informatik.

Aufgabe 2



The image shows a screenshot of a Windows command prompt window. The title bar at the top reads "z:\Classes\Informatik for D-MAVT 2010\Session 1\Example\HelloWorld\Debug\HelloWorld.exe". The window contains the following text:

```
Mein Lieblingsfach: Informatik  
Am Mar  2 2010 war mein Lieblingsfach Informatik.  
_
```