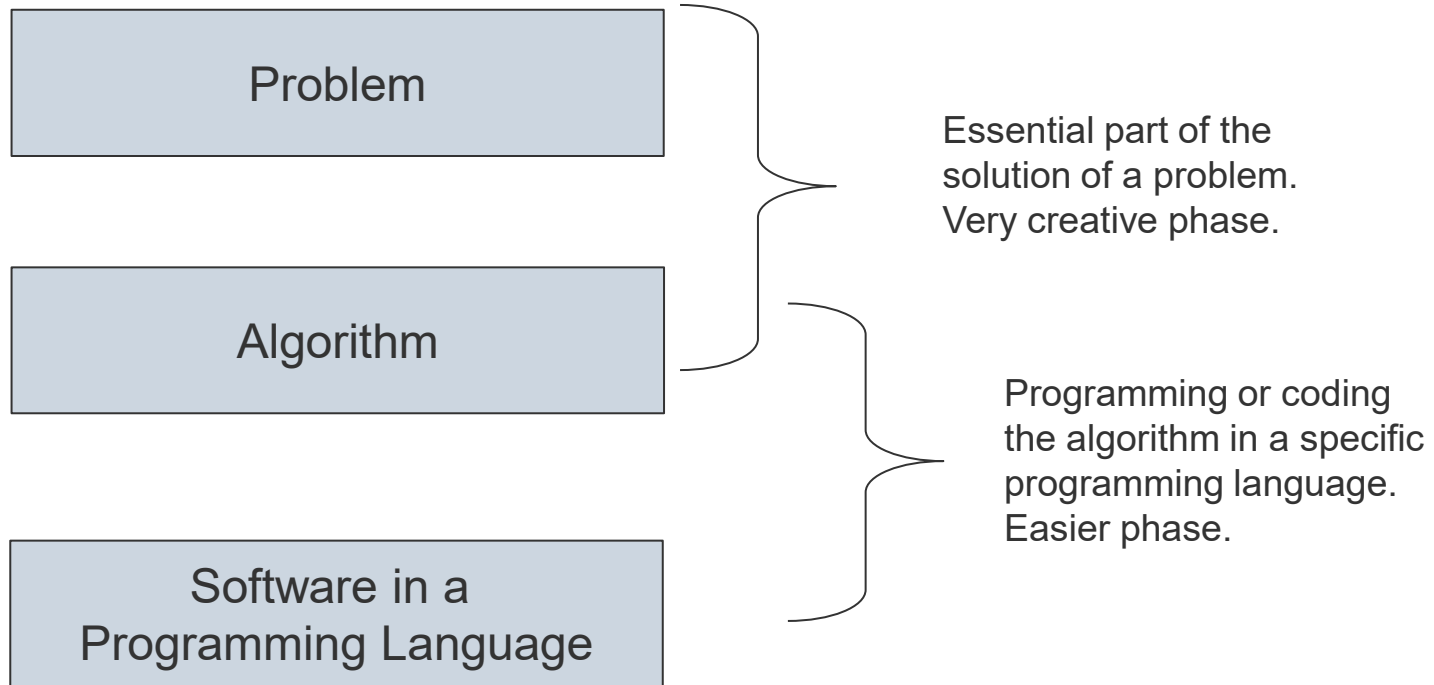


# Algorithms and Programming IV

# Recap: Concepts of Programming

Summer Term 2024 | 15.04.2024  
Claudia Müller-Birn, Barry Linnert

# Our Approach for an Efficient Solution for Problems



# Need for Higher Programming Languages

`a = b + c`



<code>mov</code>	<code>-0x8(%rbp),%eax</code>	<code>8b 45 f8</code>
<code>mov</code>	<code>-0x4(%rbp),%edx</code>	<code>8b 55 fc</code>
<code>add</code>	<code>%edx,%eax</code>	<code>01 d0</code>
<code>mov</code>	<code>%eax,-0xc(%rbp)</code>	<code>89 45 f4</code>

<code>ldr</code>	<code>r2, [fp, #-8]</code>	<code>e5 1b 20 08</code>
<code>ldr</code>	<code>r3, [fp, #-12]</code>	<code>e5 1b 30 0c</code>
<code>add</code>	<code>r3, r2, r3</code>	<code>e0 82 30 03</code>
<code>str</code>	<code>r3, [fp, #-16]</code>	<code>e5 0b 30 10</code>

# From Machine Languages to Higher Programming Languages

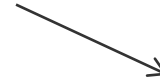
Machine Languages

Assembler Languages

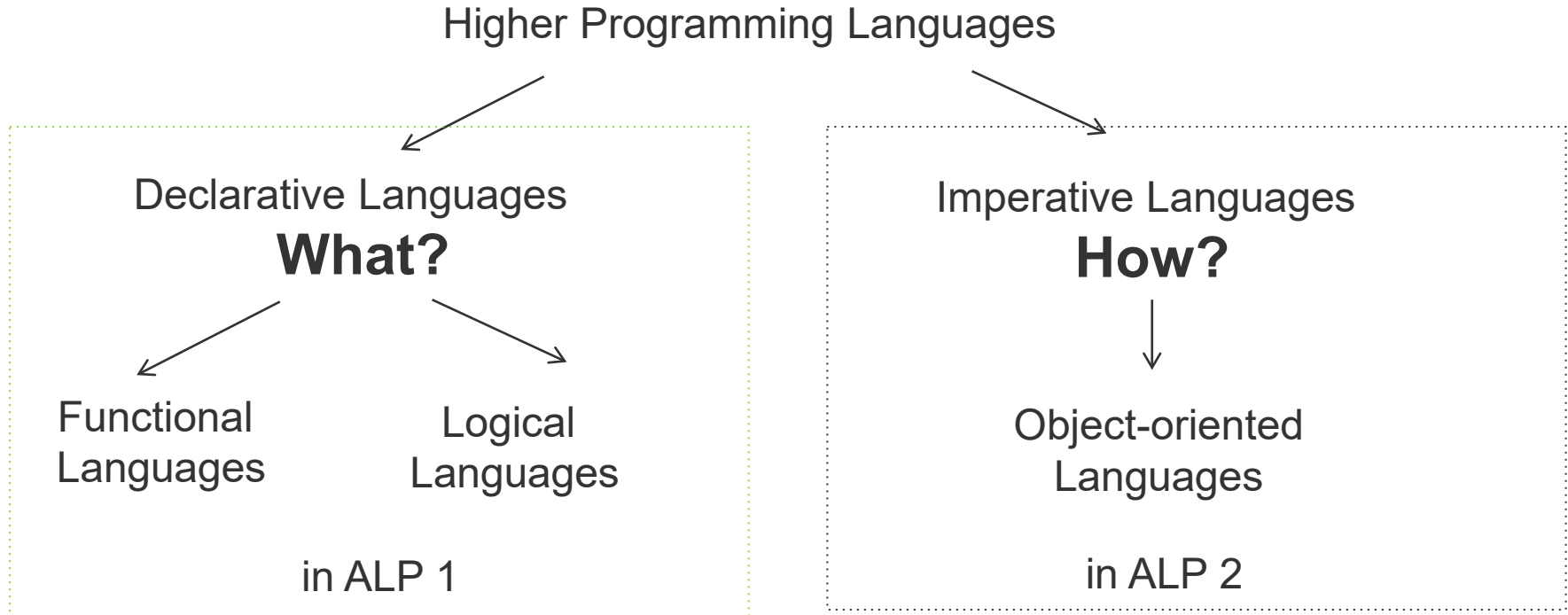
Higher Programming Languages

Declarative Languages

Imperative Languages



# Classification According to Programming Paradigm



# Classification According to Programming Paradigm

## Higher Programming Languages

**We translated an algorithm in a sequential order.**

Individual operations are executed one after the other in the sequence of instructions clearly defined in the source code. You defined thread.

Fi  
La

in ALP 1

in ALP 2

# What is a different Approach for Implementing an Algorithm?

- We can implement an algorithm non-sequentially.
- An algorithm is called non-sequential when the linear order of its operations is replaced by a non-linear order. We create multiple threads at the same time.
- However, these threads can have
  - A shared memory using one CPU -> concurrent algorithms
  - Shared or distributed memory using more than one CPU -> parallel algorithms
  - Neither shared memory nor CPU -> distributed algorithms

# Our Course Perspective

- The programming model is derived from the machine model. We explain existing mechanisms and algorithms depending on the programming model. For these, we discuss existing problems and present possible solutions.
- The course is divided into three major areas of non-sequential programming:
  - Concurrent programming: Machine with one CPU, but with a common memory
  - Parallel programming: Machine with several CPUs with or without shared memory machine, which requires message exchange
  - Distributed programming: Different machines