FORMALE SPRACHEN UND COMPILER – MODUL 1

Dominik Johannes KNOLL

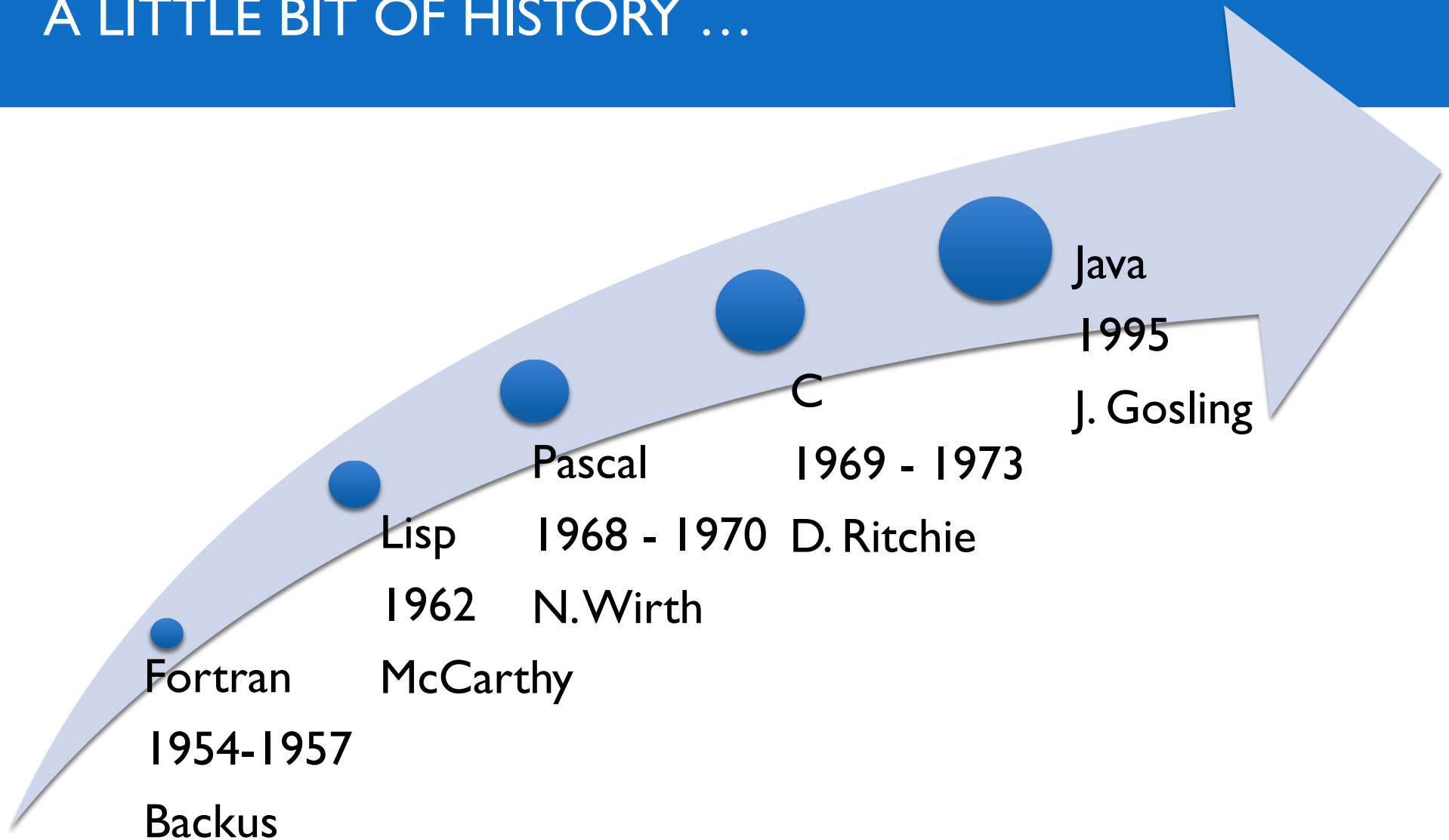
UNIVERSITATEA BABEŞ-BOLYAI
Facultatea de Matematică şi Informatică

dominik.knoll@ubbcluj.ro
djknoll.github.io

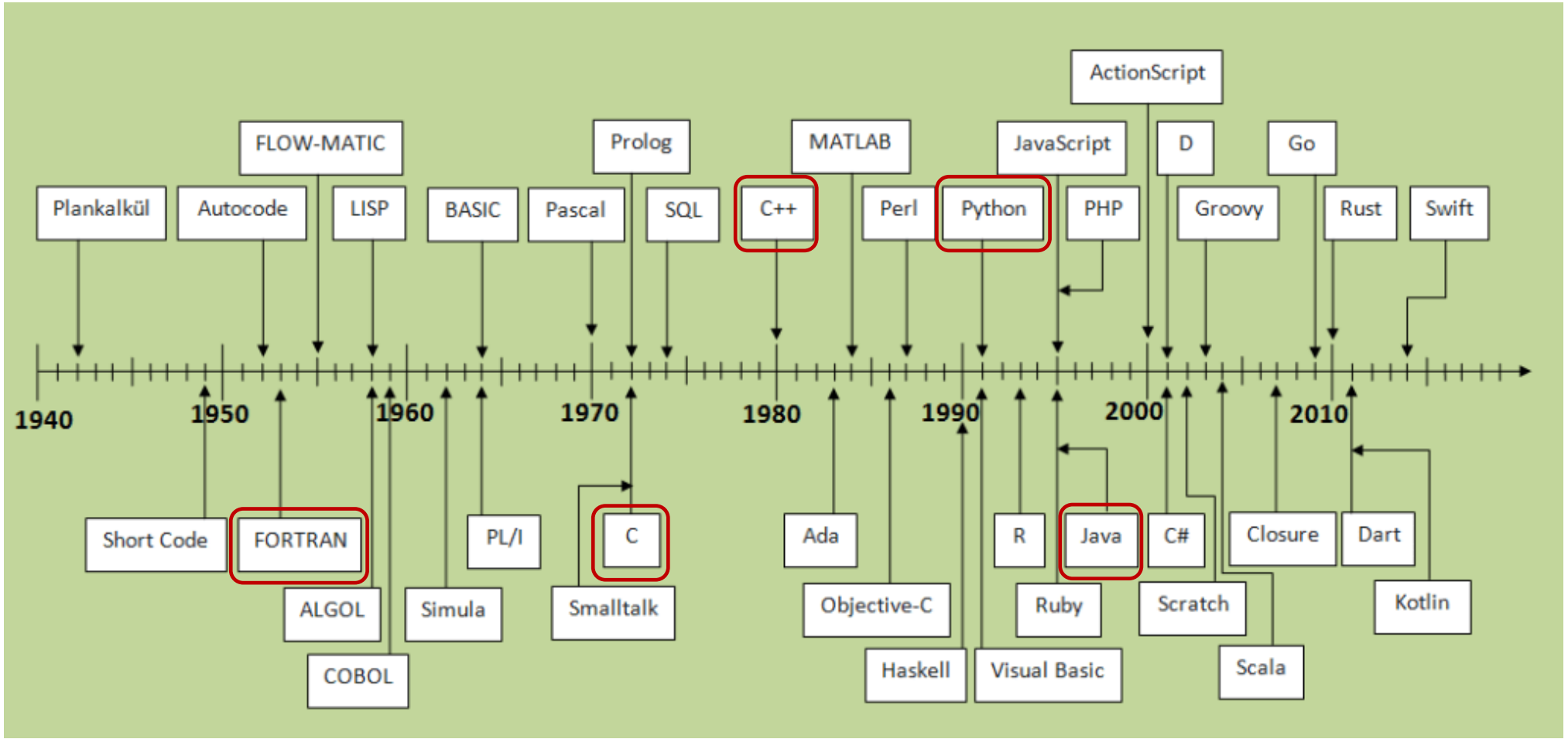
2022, v1

EINFÜHRUNG

A LITTLE BIT OF HISTORY …



A MORE DETAILED HISTORY

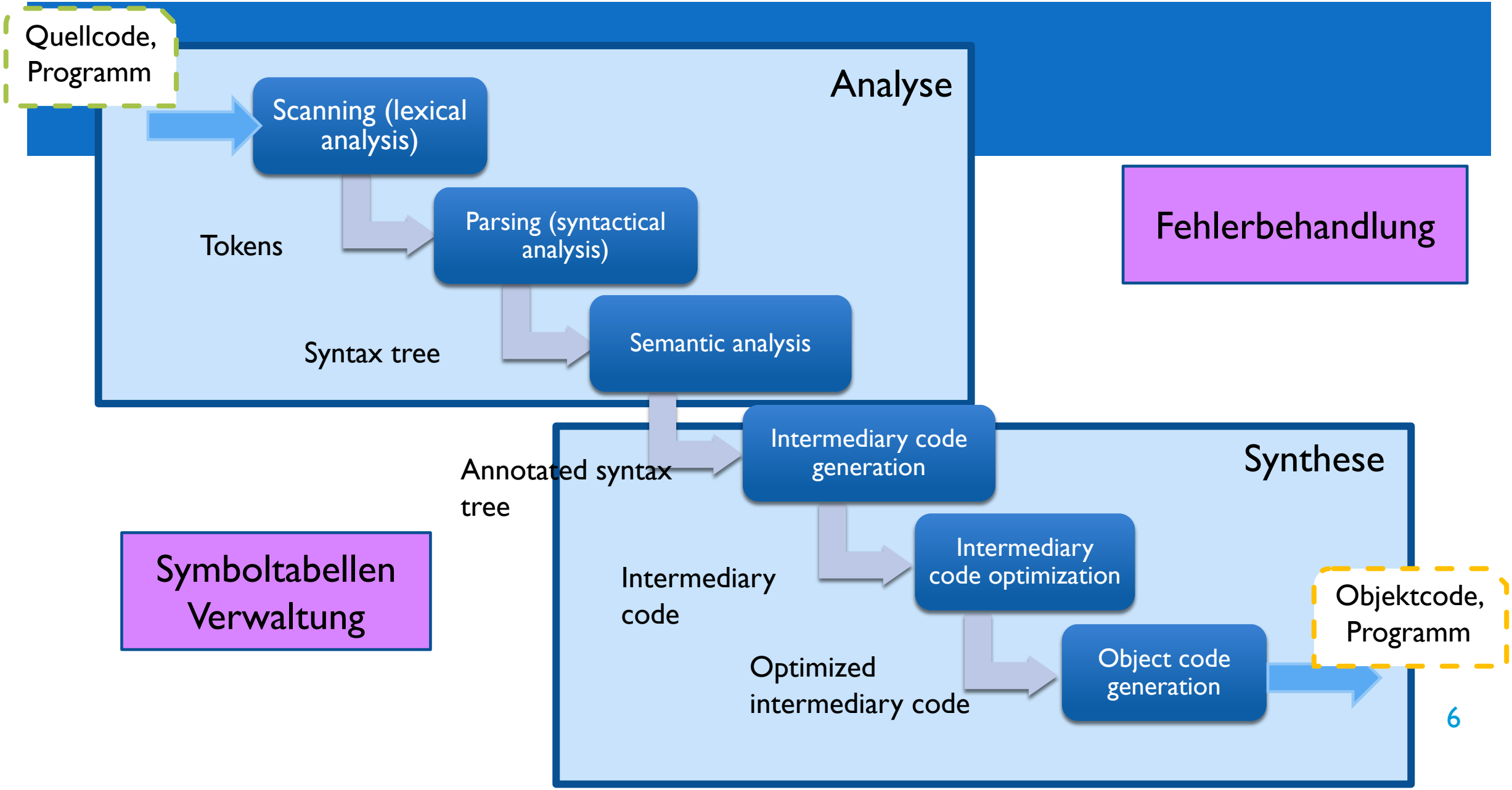


https://javaconceptoftheday.com/history-of-programming-languages/

QUIZ 1

- über Vorkenntnisse
- https://b.socrative.com/student/
- Roomname: DJKNOLL

STRUKTUR EINES COMPILERS



BEISPIELE FÜR PROGRAMMIER KONSTRUKTE

TERMINOLOGIE – BEISPIELE AUS JAVA & C++

PROGRAMMIER KONSTRUKTE I

NUMBER LITERAL

7

-94

STRING LITERAL

”Hello”

IDENTIFIER

x Y zi print

EXPRESSION

(x + 11) * 7

ASSIGNMENT

x = x + 11

PROGRAMMIER KONSTRUKTE 2

VARIABLE-DECLARATION

```
int x
```

VARIABLE-DECLARATION WITH INITIALIZATION

```
String s = "Hello"
```

CONSTANT-DECLARATION

```
const double pi = 3.1415
```

PROGRAMMIER KONSTRUKTE 3

CLASS-DECLARATION

public class Student extends Person implements Serializable BLOCK

CONSTRUCTOR-DECLARATION

public Student(String name, int age) BLOCK

CONSTRUCTOR-CALL

new Student(EXPRESSION, EXPRESSION);

PROGRAMMIER KONSTRUKTE 4

METHOD-DECLARATION

public static void main (String[] args) BLOCK

METHOD-CALL

System.out.println(“Hello UBB“);

std.cout(“Hello UBB”); || std::cout << “Hello UBB”;

PROGRAMMIER KONSTRUKTE 5

STATEMENT

if(end == true) BLOCK

switch …

while(end > start) BLOCK

for () …

return [EXPRESSION];

ASSIGNMENT ;

METHOD-CALL ;

try BLOCK catch BLOCK

BLOCK

{

statement1;

…

statementN;

}

PROGRAMMIER KONSTRUKTE 6

CONDITION,
BOOLEAN-EXPRESSION

x > 5; index <= lenth-1

EXPRESSION COMPARATOR EXPRESSION

EXPRESSION

x + 4; yi-yz+zi; x+(7-y);

NUMBER-LITERAL | IDENTIFIER | … …

”(“ EXPRESSION “)”

PROGRAMMIER KONSTRUKTE 7

FOR

```
for (int index = 0; i < lenth; i=i+ I ) { … }
for (i = start; i <= end; i++) { … }
```

FRAGE UND ANTWORTEN