PR2 - Programming 2

Lecture 2

The Java Programming Language

Outline

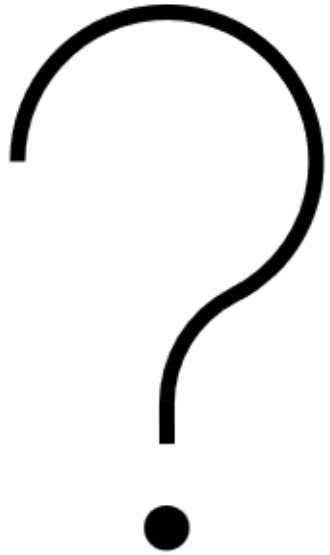
- Syntax and Semantics
- Java language syntax
- Review of Java

References

- Le Minh Duc, **Object Oriented Program Development**, Chapter 2
- J. Gosling, B. Joy, G. L. S. Jr, G. Bracha, and A. Buckley, **The Java Language Specification**, Java SE 8 Edition, 1 edition. Upper Saddle River, NJ: AddisonWesley Professional, 2014

Language Syntax & Semantics

- They eats lunch.
- They playing football.
- English is not a language.
- A ball works to earn money.



What’s wrong?

Language Syntax & Semantics

- Syntax:
  - The grammatical arrangement of words in a sentence
- Semantics:
  - The branch of linguistics that deals with the study of meaning

Programming language Syntax & Semantics

- Syntax:

```
while(1){};
int[] a = [1,2,3];
```

- Semantics:

```
String a = new String("hello");
String b = new String("hello");
If(a==b){}
```

What's wrong?

Programming language Syntax & Semantics

- Syntax:
  - The structure of statements or elements in a computer language
- Semantics:
  - what a well-formed program “means”

How to identify syntax errors?

- Syntax is defined using some form of grammar.
- Grammar is a set of rules that govern how languages elements are formed.
- With Java, we have **The Java Language Specification**, which is a book contains all rules that you need to follow when coding

How to identify semantic errors?

- Depend on your knowledge of the programming language.
- Depend on your knowledge of algorithm.

It means: you need to work hard.

Java Syntax

- Java syntax is defined using grammar rule in **The Java Language Specification**
- A grammar rule (called Context-Free Grammer) is written using this form

LHS: RHS

Where

LHS (left hand side or **nonterminal**) is a symbol or token, whose content needs to be defined

RHS (right hand side) describes the content of the LHS, which is a sequence of one or more other symbols. (more **nonterminal** and **terminal** symbols).
Terminal: pre-defined tokens

Java Syntax

- E.g. int a,b=5;

  LocalVariableDeclarationStatement:
  LocalVariableDeclaration ;

  LocalVariableDeclaration:
  {VariableModifier} UnannType VariableDeclaratorList

Grammar Notation

- Concatenation: x y …
  - sequentially listing the elements, separated by spaces (‘ ’)
- Selection: (one of) x y ...
- Option: [x]
  - Zero or one occurrences of x
- Repetition: {x}
  - zero or more occurrences of x

Question

- Can we represent a number by using 1_0?
- What is the value of `1_0` in java?

What does this rule mean?

DecimalNumeral:

0

NonZeroDigit [Digits]

NonZeroDigit Underscores Digits

Java Syntax

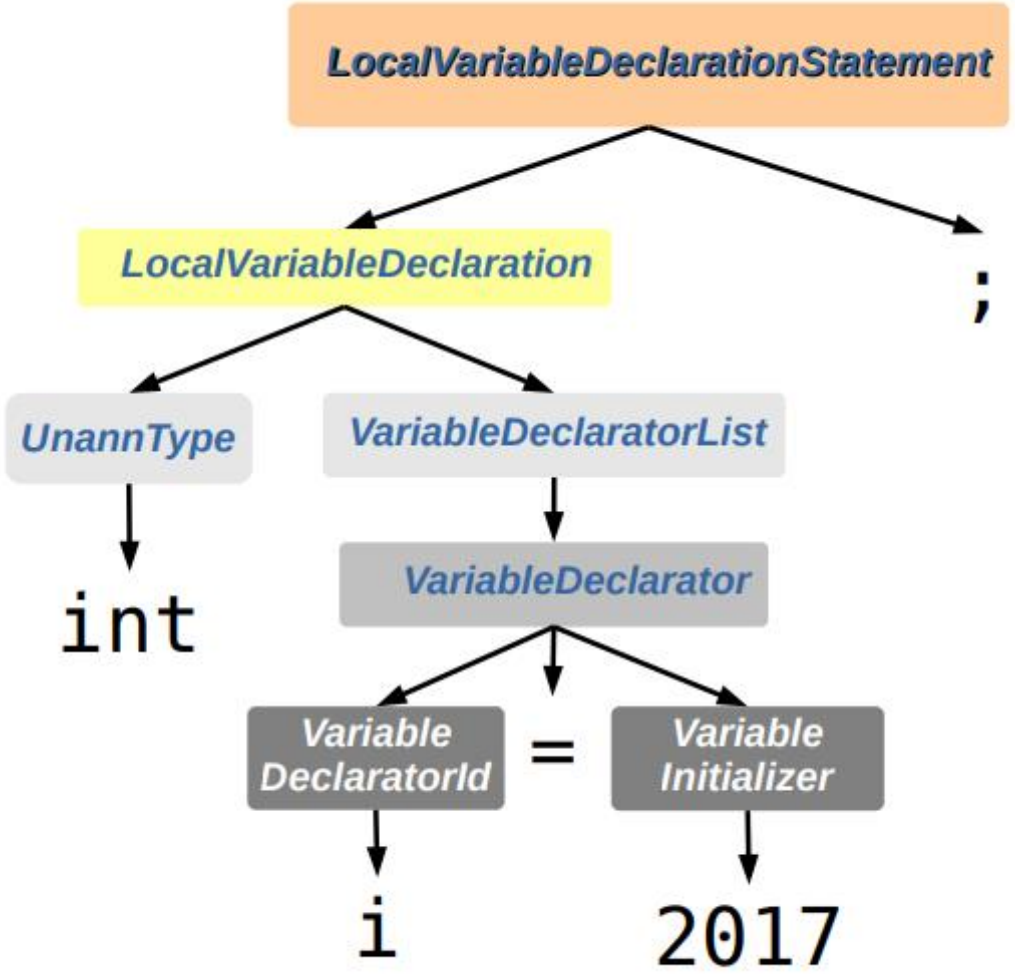
Represent by using:

- Syntax Tree
- Nested Block Diagram

Syntax Tree

- Code statement:

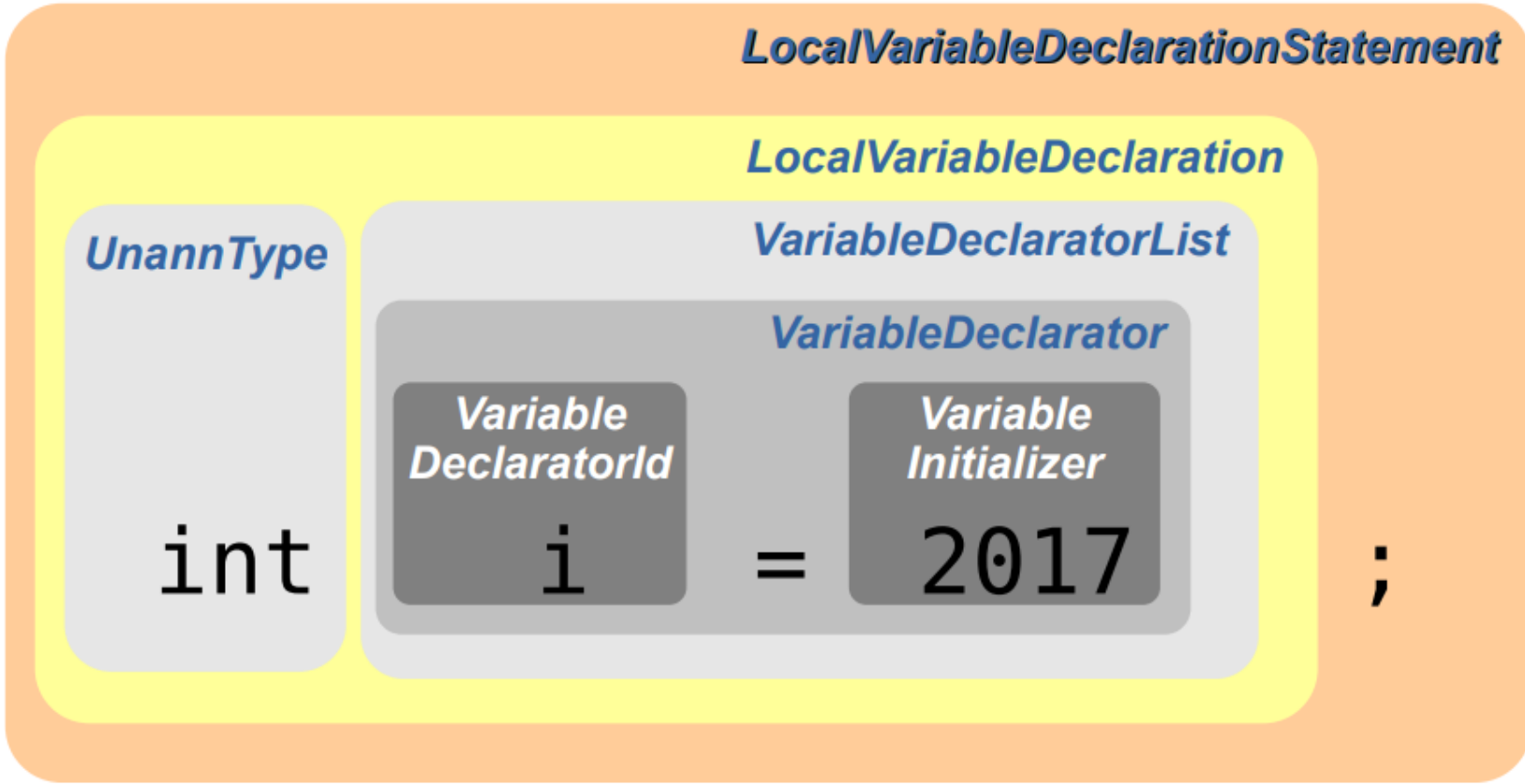
int i = 2017;



Nested Block Diagram (NBD)

- Code statement:

int i = 2017;



Java language syntax

In **The Java Language Specification**:

- Identifier (Chapter 3.8)
- Method Declaration (Chapter 8.4)
- Blocks and statements (Chapter 14)
- Expressions (Chapter 15)

Examples

- ClassDeclaration:
- Method declaration:
- LocalVariableDeclarationStatement:
- Statement:
- IfStatement:
- ForStatement:
- WhileStatement:
- SwitchStatement:

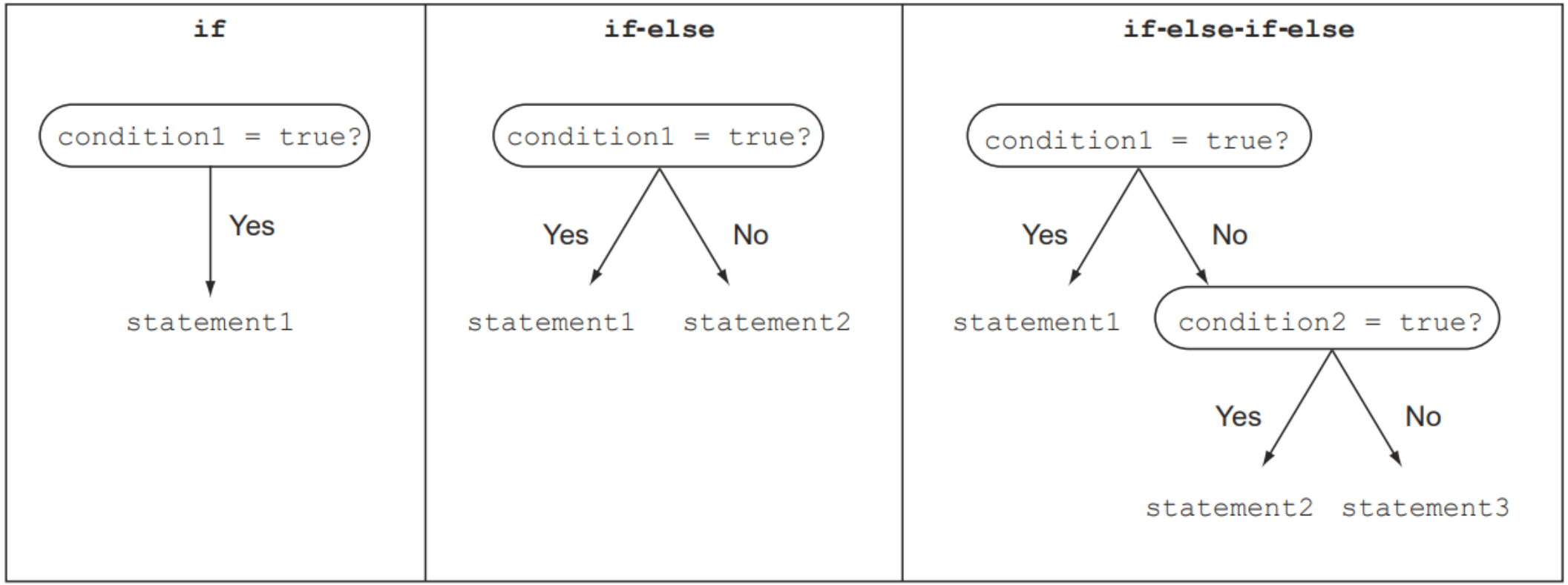
Review Java

- Branch statement
- Switch
- Loop
- Labels statement

Branch statements

- If
- If-else
- If-else-if-else
- Nested if

Branch statements



Questions: What is the results?

```
int score = 50;
if((score=score+10)==70){
   System.out.print("A:");
} else if ((score=score+20)==90){
   System.out.print("B:");
} else {
   System.out.print("C:");
}
System.out.println(score);
```

```
boolean test = false;
if(test){
} else
   System.out.println("hi");
```

```
boolean test = false;
if(test)
else
   System.out.println("hi");
```

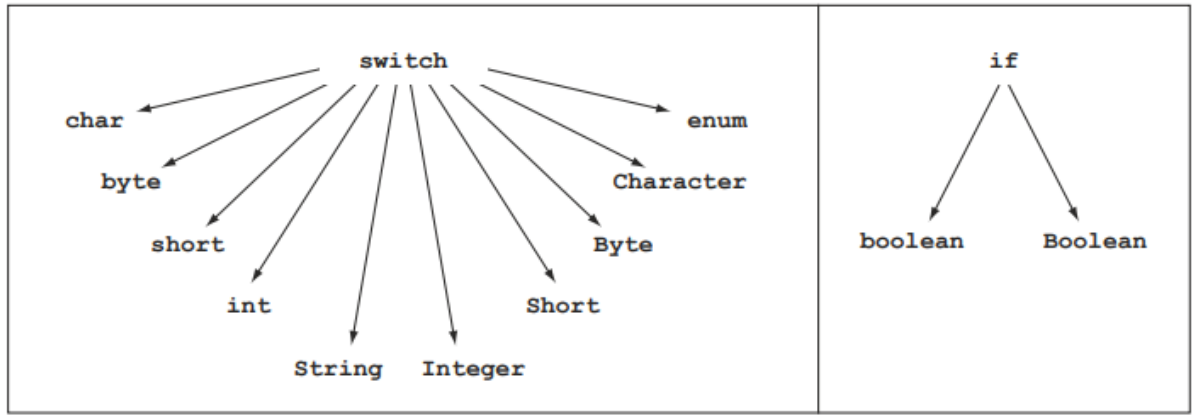
```
boolean test = false;
if(test)
   System.out.println("hi");
   System.out.println("hi");
```

Switch statement

```
switch (expression) {
     case constant1: code block
     case constant2: code block
     default: code block
}
```

Switch vs if-else-if-else

- Type of argument passed to switch statement
- Switch statement can improve the readability of your code.



Questions: Switch statement

```
int score = 5;
switch (score+4) {
      case 5: result = "A";
      case 9: result = "B";
      default: result = "C";
}
//??? result
```

```
float score = 5;
Switch (score+4) {
      case 5: result = "A";
              break;
      case 9: result = "B";
              break;
      default: result = "C";
}
//??? result
```

Loop

- for
- while

The for loop

```
for(initialize; condition; update){
      statements;
}
```

```
for(; condition; update){
      statements;
}
```

```
for(initialize; condition;){
      statements;
}
```

```
for(initialize; ; update){
      statements;
}
```

```
for(; ;){
      statements;
}
```

The while and do-while loop

```
while(condition){
        statements;
}
```

```
do {
        statements;
} while(condition);
```

For loop vs while loop

- Should use for loop when you know the number of iterations
- Should use while loop when you don’t know the number of iterations

Break and continue statement

- break statement is used to exit the loop.
- continue statement is used to skip the remaining steps in the current iteration and start with the next loop iteration.

Labels statements

```
int[] scores = {1,2,3};
      for(int score1:scores){
          for(int score2:scores){
            if(score2==2)
                  //how to exit outer loop???
          }
      }
```

Labels statements

```
int[] scores = {1,2,3};
outer:
for(int score1:scores){
      for(int score2:scores){
            if(score2==2)
                  break outer;
      }

}
```

Labels statements

- In java, you can add labels to the following types of statements:
  - Code block defined using {}
  - All loop statement
  - Conditional construct
  - Expression
  - Assignment
  - Return statement
  - Try block
  - Throws statement

Summary

- The difference between syntax and semantic:
  - Syntax is the grammatical arrangement of words in a sentence
  - Semantic is what a well-formed program “means”
- Grammar rule (called Context-Free Grammer) in java syntax:
  - LHS: RHS
- Example of java language syntax and using Syntax-Tree or Nested Block Diagram to describe them
- Review of Java