ETH zürich

Assignment 1

Felix Friedrich, Lars Widmer

TA lecture, *Informatics II D-BAUG*

March 11, 2014

Introduction

- Name
- Years/semesters at ETH?
- Home Town
- Mail address
- Mobile number
- “Work field”
  - Research topic
  - Master thesis
  - Bachelor thesis
- Hobbies
- . . .

Eclipse Installation

- Live Demo
- By request . . .
- On a students laptop . . .

Java Installation

- Live Demo
- By request . . .
- On a students laptop . . .

Hello World Program

- Live Programming
- By request . . .

Matlab vs. Java

Would you use Matlab or Java for the following tasks?

1. Larger project which code will be modified and reused.
2. Complex matrix computations for your masters thesis.
3. Project with 7 people working on at the same time.
4. Free software that everybody should be able to download and run.
5. Software which can be used intuituivly, with a nice GUI.
6. Program for a mobile phone.
7. Object oriented code project.

Matlab vs. Java

Would you use Matlab or Java for the following tasks?

1. Larger project which code will be modified and reused.
   - Java offers better reusability.
2. Complex matrix multiplications for your masters thesis.
   - Easier with Matlab because of a huge set of prepared libraries
3. Project with 7 people working on at the same time.
   - Usually better with java
4. Free software that everybody should be able to download and run.
   - Matlab is proprietary, you have to use java for free software.

Matlab vs. Java

Would you use Matlab or Java for the following tasks?

5. Software which can be used intuituivly, with a nice GUI.
   - Intuitive software is easier to achieve in java.
6. Program for a mobile phone.
   - There's no Matlab environment on mobile phones, while java is supported very often.
7. Object oriented code project.
   - Java is object oriented which aids code structure, readability and reusability.

Main Memory

1. What is Main Memory?
2. Whats a bit?
3. Whats a byte?
4. How many states can a byte have?
5. What is main memory used for?
6. How do you access it?

Main Memory

1. What is Main Memory?
   - A series of bits of storage
2. Whats a bit?
   - A single bistable storage cell
   - It is either `0` (`false`) or `1` (`true`)
3. Whats a byte?
   - A series of 8bits

Main Memory

1. How many states can a byte have?
   - A byte can have 256 different states
   - 00000000, 00000001, 00000010, ... 11111111
2. What is main memory used for?
   - It represents the state of a program.
3. How do you access it?
   - Every storage cell (e. g. 1Byte) has its own address.

Processor

1. What does a processor?
2. Name an example of a processor instruction

Processor

1. What does a processor?
   - It runs through the machine code of a program and fulfills these small steps of work.
2. Name an example of a processor instruction:
   - Load a value from main memory into a register
   - Add two registers and store the result in a register
   - ...

Java, Virtual Machine

1. Is the code of a java program platform dependent?
2. And a compiled java program?
3. How does the virtual machine work?

Java, Virtual Machine

1. Is the code of a java program platform dependent?
   - Nope, it runs on a virtual machine (VM).
   - The VM takes the same code on every platform (Linux, Mac, Win, Mobile, . . .)
2. And a compiled java program?
   - Java programs are compiled to bytecode (sort of a generic machine code).
   - Bytecode is still platform independent.
3. How does the virtual machine work?
   - Basically it simulates a generic processor.

Precedence & Associativity

Translate the following into pascal code into java:

```
1 IF (i<>0) AND (j DIV j = 10) THEN
2 BEGIN
3       ....
4 END;
```

Precedence & Associativity

Pascal:

```
1 IF (i<>0) AND (j DIV j = 10) THEN
2 BEGIN
3       ...
4 END;
```

Java:

```
1 if (i != 0 && j/i == 10)
2 {
3       ...
4 }
```

Precedence & Associativity

```
1 if (i != 0 && j/i == 10) {...}
```

- Defined by the syntax, java needs brackets after the “`if`”.
- Inside of the brackets precedence defines the order at which statements are evaluated.

Precedence & Associativity

These two lines give the same result.

```
1 if (i != 0 && j/i == 10) {...}
```

```
1 if ((i != 0) && ((j/i) == 10)) {...}
```

Commutativity

- Addition in java is not always commutative.
- Addition for `Strings` results in a **concatination**, which is **not commutative**.
- When adding `ints` to `Strings`, java converts the numbers into `Strings`
- What's the result (`str1, str2, str3`) of the following assignments?

```
String str1 = "a " + 10;
String str2 = 10 + "a";
String str3 = "a " + 10 + 20;
```

Commutativity

```
String str1 = "a " + 10;
String str2 = 10 + "a";
String str3 = "a " + 10 + 20;
```

- `str1` becomes “a 10”
- `str2` becomes “10 a”
  `str1` is clearly not equal to `str2`
- `str3` becomes “a 1020”
  Therefore addition of strings is **left associative**

Increment & Decrement

- Especially in for-loops it's common to use a short form of **increment** and **decrement**.
- Feel free to always use the simple and clear forms like "`i = i + 1;`". The speed gain is usually extremly small or even non existant.

Pascal:

```
INC(i); // increment: i := i + 1;
DEC(i); // decrement: i := i - 1;
```

Increment & Decrement

Pascal:

```
INC(i); // increment: i := i + 1;
DEC(i); // decrement: i := i - 1;
```

Java:

```
int a = ++i // increment and return new value:
i = i + 1; int a = i; // same result

int a = i++ // increment and return old value:
int a = i; i = i + 1; // same result
```

Any questions?

- Feedback?
- Wishes?
- Hopes?
- ...

Good Luck