

Introduction to Programming – Concepts and Tools

Carsten Butz
IT University of Copenhagen

Week 1

Today's Goals

- Introduction
- Practical Information
- Course Outline
- Lecture:

Carsten Butz

1

Practical Information

- Course: Friday, 9:00 – 12:00 in room 0.10
 - Exercise classes/Labs: Friday, 13:00 – 16:00 in rooms 3.19, 4.04, 4.05
 - Lecturer: Carsten Butz, office 1.17, email: butz@it-c.dk
 - Office hours: Thursday, 12:00–13:00, Friday, 16:00–17:00
 - Teaching assistants:
 - Kristian Stengaard Munkholm
 - Lars Rune Christensen
 - ?
 - Course homepage: www.it-c.dk/courses/IPBR/
 - Usually updated Thursday evening before the course
- All material will be posted on the webpage

Carsten Butz

2

Practical Information

- Assignments: 11 Assignment sheets of which at least 8 have to be completed successfully to be allowed to sit the exam.
 - Assignments have to be handed in on Fridays before the end of the lecture (12:00).
 - The written part of the assignment has to be stapled, with the front page being the cover sheet to be found at the coursepage.
 - The code has to be submitted using the `submit` command under Linux. For details see the first or second assignment sheet.

Carsten Butz

3

Practical Information

- Exam: January 3rd, 2003.
 - Open book (i.e., you are allowed to bring your textbook and notes to the exam room)
 - Language dictionaries are allowed
 - No electronic devices are allowed
 - Graded on the Danish 13 scale

Carsten Butz

4

Practical Information

- Workload:
 - 36 hours of lectures
 - 36 hours of lab sessions with lab assistants
 - 4 hours exam
 - 74 hours of self study

Carsten Butz

5



Project Period

- When: Nov. 18 – Dec. 20 (12:00)
- What: Larger programming projects in groups of 2 to 5 students.
- Value: 7.5 ECTS
- Examples:
 - Search engine
 - Servlet/Database applications like
 - Online registration of student's course wishes
 - Simple game with ranking system
 - Webbased discussion forum
 - Analysis of server log-files
 - Etc.



Textbook and Other Resources

- Textbook: Pohl/McDowell, Java by Dissection. The essentials of Java Programming. Addison-Wesley, 2000 or later.
- Material posted on the course page, including
 - Slides from the lectures
 - Assignment sheets
- Internet resources



Why English?

- There are foreign students here who don't speak Danish.
- English is the lingua franca for Information Technology and Internet Technology.
- Many potential employers expect that you are fluent in English.
- Text, documents, descriptions, etc. both in written form and on the internet of often in English, and English only.
- ...
- ... your lecturer might not speak Danish !



Classroom Etiquette

- Be concentrated, come on time (as do I).
- Don't disturb the atmosphere by talking to you neighbour.
- Questions are allowed, preferably in English, but Danish is fine.
- However, I may not be able to answer all questions, and may refer you to the text.



Your Teacher

- Native German
- Master's in Mathematics (minor in Computer Science)
- Ph.D. in Mathematics from Utrecht University (The Netherlands)
- Researcher and Lecturer in Aarhus, Montreal, Edinburgh before coming to Copenhagen
- Research area: Foundations of mathematics and computer science, logic, semantics of programming languages.



Contents

- Introduction to concepts standard to most common programming and scripting languages
 - Programming process
 - Variables, identifiers
 - Control structures
 - Data types
 - Algorithms and algorithm analysis (searching, sorting)
 - Introduction to object oriented programming (inheritance, encapsulation, polymorphism)
 - Exception handling
 - Graphical user interfaces (if time permits)
- All this is (programming) language independent!



Contents

- Introduction to a high level programming language: Java
 - How the concepts are realised in Java
 - How to write programs using Java



Motivation

- Why learning programming?
 - Needed skill for all technical courses
 - The only way to understand possibilities and limitation of information technology
 - Helps understand potential or new products or solutions
 - Vital skill for the job market
 - Full of transferable skills
- Why Java?
 - Solid high level object oriented programming language
 - Well suited for internet applications and user interfaces
 - Good language for teaching abstracting away some technical details



Programming language levels

- Machine language
 - Specific to each CPU (chip)
 - Expressed using binary digits
 - Extremely simple tasks
- Assembly language
 - Replace binary digits by short English-like mnemonics
 - Programs must first be translated into machine code
- High-level language
 - English like phrases
 - Simple commands accomplish complex tasks
 - Machine independent
- Fourth generation languages (4GL)

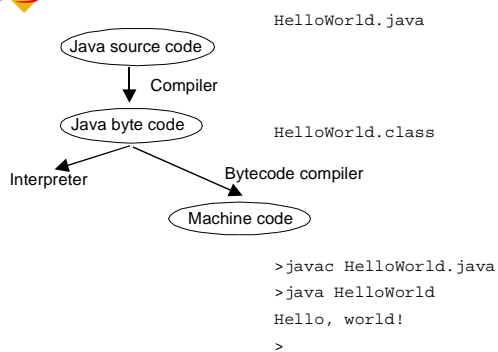


Compiler and Interpreter

- Compiler: Program that translates from one language into another
- Interpreter: Program that executes a program in a language different than machine language and interprets while reading the source code. Thus, translation and execution is done at the same time.
 - Usually slower than first compiling into machine code, and then running the machine code.



Java translation and execution process



HelloWorld.java

```

/* HelloWorld.java
 * Purpose:
 * The classic "Hello, world!" program.
 * It prints a message to the screen.
 * Author: Jane Programmer
 * as derived from Kernighan and Richie
 */
class HelloWorld {
    public static void main (String[] args) {
        System.out.println("Hello, world!");
    }
}
  
```

Comments (author, date, purpose)
 use // per line or /* ... */
 ← def of class HelloWorld
 ← def of main method
 ← body of main method



Compilation process

- Create the file `HelloWorld.java` with a text editor.
 - Preferred choice is `emacs`
- Compile the program by typing `javac HelloWorld.java`
- Run the program by typing `java HelloWorld`

■ On the screen the last two steps look as

```
>javac HelloWorld.java
>java HelloWorld
Hello, world!
>
```



A note about syntax

- Syntax is not the important thing, you have to understand the concepts.
- If you do not understand the concepts you will not be able to program.

However,

- To get your program compiling it is vital that the syntax is correct.
- Without correct syntax your program will not run.
- Finding your own (syntax) errors will unfortunately be one of your prime tasks.