Java across Different Curricula, Courses and Countries Using a Common Pool of Teaching Material

Mirjana Ivanović, Zoran Budimac, Anastas Mišev, Klaus Bothe, Ioan Jurca

DAAD project
"Joint Course on OOP using Java"
Humboldt University Berlin, University of Novi Sad, Polytehnica University of Timisoara, University of Plovdiv, University of Belgrade

Agenda

- Final selection of topics up to this year
  - Basic topics
  - Advanced topics

- Changes caused by introducing OOP II course

- Conclusion
Final Selection of Topics

- Basic topics
  - Cover basic Java programming concepts
  - Intended for students who may or may not already be familiar with OOP concepts

- Advanced topics
  - Cover many advanced Java topics
  - Intended for students who wish to broaden their knowledge
  - Students must be familiar with basic Java programming concepts
  - Mainly for elective or higher-year courses

Possible usage of available topics in 6 different courses at 2 universities

- Object-Oriented Programming I
- Data Structures and Algorithms
- Object-Oriented Programming II
- Network Operating Systems
- Data Bases II
- Operating Systems I
Agenda

- Final selection of topics up to this year
  - Basic topics
  - Advanced topics

- Changes caused by introducing OOP II course

- Conclusion

Changes caused by introducing OOP II course

- Object-Oriented Programming II (elective course)
- Novi Sad, 4th semester
- Originally planned topics

<table>
<thead>
<tr>
<th>Topic</th>
<th>Presentation</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>T12. Strings</td>
<td></td>
<td>String, StringBuilder, StringBuffer, StringTokenizer, Regular Expressions, Formatting Input, Scanning Output</td>
</tr>
<tr>
<td>T13. Windows &amp; Applets</td>
<td></td>
<td>Detailed version</td>
</tr>
<tr>
<td>T14. Collections</td>
<td></td>
<td>Arrays, containers, generics</td>
</tr>
<tr>
<td>T15. The Java I/O System</td>
<td></td>
<td></td>
</tr>
<tr>
<td>T16. Serialization</td>
<td></td>
<td></td>
</tr>
<tr>
<td>T17. Java 5 New Features</td>
<td></td>
<td>Generics, boxing, varargs, enhanced for loop, enumerations, static imports, annotations, formatting, threading, overriding return types, unicode</td>
</tr>
<tr>
<td>T18. Generics</td>
<td></td>
<td></td>
</tr>
<tr>
<td>T19. Enumerated Types</td>
<td></td>
<td></td>
</tr>
<tr>
<td>T20. Annotations</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
**Changes caused by introducing OOP II course**

1. In order to fit the course into the 12-week semester, the topics on **Java security and JDBC were excluded**
   - JDBC is studied extensively in other courses

2. **Mobile agents** were merged with the distributed programming topic, as a case-study

3. **GUI programming** was split into two separate topics – one covering **advanced Swing components**, and the other focusing on customizing **Swing-based applications**

4. **Java internet programming and Enterprise JavaBeans topics** were re-organized as a two-week long Java EE topic

5. Finally, a topic on **OO design (UML and Design patterns)** was introduced
Changes caused by introducing OOP II course - Modifications of individual topics (1/3)

- Many topics were updated in accordance to recent developments of the Java technology

- **Strings:**
  - The *split* method was studied instead of *StringTokenizer*, as encouraged by Java documentation
  - Input and output were moved to topic on Java I/O

- The Java I/O system was extended with *NIO*, since this package is important for **advanced network programming**

Changes caused by introducing OOP II course - Modifications of individual topics (2/3)

- **Threads:**
  - The concept of *Timers* was included
  - Only the *Dining Philosophers* problem was used as a **classical synchronization problem**

- The topic on **network programming was extended with multiplexing**

- **Distributed programming** now also includes an overview of CORBA, Reflection API, and class-loaders
Changes caused by introducing OOP II course - Modifications of individual topics (3/3)

- Java EE topic was introduced as the merger of Java internet programming and Enterprise JavaBeans topics proposed initially.

- Additional changes, following the recent Java developments, include:
  - Asynchronous Enterprise JavaBeans invocation
  - Java Persistence API instead of Entity beans
  - Java Server Faces instead of Java Server Pages
  - SOA-based architectures and web services as a new content of the topic

- Servlets were not studied due to the lack of time.

Agenda

- Final selection of topics up to this year
  - Basic topics
  - Advanced topics

- Changes caused by introducing OOP II course

- Conclusion
Conclusion

- The proposed organization of the teaching material is very important because:
  - it covers a wide variety of Java topics,
  - it can be used for different courses,
  - different topics can be used in different institutions (project members).

- The material is well prepared, significantly innovated within OOP II course
  - Good bases for further innovations of teaching material for advanced topics.

- Constant improvement, modernization, and expansion of the material.

Java across Different Curricula, Courses and Countries Using a Common Pool of Teaching Material

Mirjana Ivanović, Zoran Budimac, Anastas Milișev, Klaus Bothe, Ioan Jurca

DAAD project “Joint Course on OOP using Java”

Humboldt University Berlin, University of Novi Sad, ‘Polytehnica’ University of Timisoara, University of Plovdiv, University of Belgrade