Spring 2005

CS 884: Advanced Topics in Programming Languages

Krishnaprasad Thirunarayan
Wright State University - Main Campus, t.k.prasad@wright.edu

Follow this and additional works at: https://corescholar.libraries.wright.edu/cecs_syllabi

Part of the Computer Engineering Commons, and the Computer Sciences Commons

Repository Citation

This Syllabus is brought to you for free and open access by the College of Engineering & Computer Science at CORE Scholar. It has been accepted for inclusion in Computer Science & Engineering Syllabi by an authorized administrator of CORE Scholar. For more information, please contact corescholar@www.libraries.wright.edu, library-corescholar@wright.edu.
CS 884 Advanced Topics in Programming Languages

- **Instructor:** T. K. Prasad  
  **Phone No.:** (937)-775-5109  
  **Email:** t.k.prasad@wright.edu  
  **Home Page:** [http://www.cs.wright.edu/~tkprasad](http://www.cs.wright.edu/~tkprasad)

- **Quarter:** Spring, 2005  
  **Class Hrs:** Tu Th, 6:05 to 7:20pm, 134 Health Sciences  
  **Office Hrs:** Tu Th, 5:30 to 6pm and 7:30 to 8pm, 337 RC (or by appointment)

---

**Course Objectives**

- To analyze, design, and specify modern programming languages.

---

**Prerequisite**

- [CS 784 Programming Languages OR CS 780 Compiler Design and Construction I](#)

---

**Course Description**

The primary focus of this course is the design and specification of the Object-Oriented language Java.

---

**Course Load**

The course load includes a term-paper and presentation worth 15 points, programming assignments worth 25 points, a midterm worth 30 points, and a final worth 30 points. Exams are typically open book.

---

**Texts**


---

**References**

Relevant Websites

- [Java 5 Core APIs](#)
- [The Java Tutorial](#)
- [Research on Java Implementation](#)
- [Java Tools](#)

---

**Grading**

The letter grades will be assigned using the following scale: A[90-100], B[80-90), C[70-80), D[60-70), and F[0-60). However, I reserve the right to adjust the scale somewhat to utilize the gaps in the distribution.

**Attendance Policy**

All registered students are expected to attend all lectures. In case a student is absent from a lecture due to unavoidable circumstances, the student is still responsible for the material covered in the class, as it is typically available from the course web-page well in advance. Furthermore, the student is expected to find out about in-class announcements from their colleagues/instructor.

---

**Class Schedule and Syllabus**

<table>
<thead>
<tr>
<th>Topic</th>
<th>Class 0</th>
<th>Class 1</th>
<th>Class 2</th>
<th>Class 3</th>
<th>Class 4</th>
<th>Class 5</th>
<th>Class 6</th>
<th>Class 7</th>
<th>Class 8</th>
<th>Class 9</th>
<th>Class 10</th>
<th>Class 11</th>
<th>Class 12</th>
<th>Class 13</th>
<th>Class 14</th>
<th>Class 15</th>
<th>Class 16</th>
<th>Class 17</th>
<th>Class 18</th>
<th>Class 19</th>
</tr>
</thead>
<tbody>
<tr>
<td>The Aesthetics of Simplicity</td>
<td><a href="#">The Aesthetics of Simplicity</a></td>
<td><a href="#">Motivation for Formal Semantics</a></td>
<td><a href="#">Java: Design Goals</a></td>
<td><a href="#">Java Constructs</a>  ; <a href="#">Examples</a></td>
<td><a href="#">Values, Variables, and Types</a></td>
<td>(cont'd)</td>
<td>Names : Scope, Access; Packages</td>
<td>(cont'd)</td>
<td>Classes : Inheritance, Polymorphism</td>
<td>(cont'd)</td>
<td>Midterm (April 28)</td>
<td><a href="#">Interfaces; Exceptions</a></td>
<td>Expressions; Statements; Finalization</td>
<td><a href="#">Concurrent Programming with Threads</a></td>
<td>(Oak IR <a href="#">ps</a> [pdf])</td>
<td><a href="#">GC</a></td>
<td>Inner classes and Reflection ; Examples</td>
<td>Presentation</td>
<td>Presentation</td>
<td></td>
</tr>
</tbody>
</table>
Class 20  Presentation  
Finals (June 9, 8 - 10pm)

Assignments (Spring 05)

- Assignment 1.
- Assignment 2.

Exams (Spring 03)

- Midterm.
- Final.

T. K. Prasad  (21 Mar 2005)