Spring 2000

CMSC 111: Structured Programming I

Marv Wielard  
Dordt College

Follow this and additional works at: https://digitalcollections.dordt.edu/spring2000

Part of the Computer Sciences Commons

Recommended Citation

This Syllabi is brought to you for free and open access by the 2000 at Digital Collections @ Dordt. It has been accepted for inclusion in Spring 2000 by an authorized administrator of Digital Collections @ Dordt. For more information, please contact ingrid.mulder@dordt.edu.
A. Course Title: Structured Programming I

B. Course Description
This course is designed to be an introduction to structured computer programming using the Java programming language. Program structure and style are major concerns. There is also an emphasis on problem solving; in particular, the following four steps are stressed: 1) problem analysis, 2) solution design through stepwise refinement, 3) computer implementation, and 4) testing.

C. Course Objectives
Upon completion of this course, the student will:

1) relate basic themes of redemptive history to a Christian perspective on computer science
2) relate basic normative principles to programming activities
3) know the basic components of a computer system
4) know the basic syntax and reserved words of the Java programming language
5) be able to give expression to the requirements of a programming problem, use top-down design techniques to create a well structured algorithm, create an adequate test plan, express the algorithm in Java, implement and test the Java program incrementally.
6) be able to use effectively the programmer's basic tools: variables; programming structures: sequence, 3 types of selection, 3 types of repetition; methods (modules); and arrays.
7) advance in assuming responsibility for his/her own learning
8) improve reading, critical-thinking, and problem-solving skills

D. Methods of Instruction
Instruction will take place through readings and exercises in the text, through class discussions, and through programming exercises and projects.

E. Major Assignments and Projects
There is no major project. But there are almost daily programming exercises and projects. Some small group projects may be assigned.

F. Evaluation
Quizzes, assignments, class participation, and projects 40%
Three tests 45%
Cumulative Exam 15%

Grading Scale: 90's = A; 80's = B; 70's = C; 60's = D

Tests and examination must be taken in class on the date indicated by the instructor unless a different arrangement is made prior to test administration.

G. Text
H. Instructor
Name: M. Wielard Office: CC02
Extension: 6298 E-mail: wielard@dordt.edu
Office Hours: Mon, Wed, and Fri 1:00 - 2:00; Tue 2:00 - 3:00

I. Course Schedule

<table>
<thead>
<tr>
<th>Chapter Number</th>
<th>Chapter Name/Topic</th>
<th>Class Periods</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Intro to Computers</td>
<td>1-8</td>
</tr>
<tr>
<td>2</td>
<td>Intro to Java Applications</td>
<td>1-8</td>
</tr>
<tr>
<td>3</td>
<td>Intro to Java Applets</td>
<td>9-13</td>
</tr>
<tr>
<td></td>
<td>Test One</td>
<td>14</td>
</tr>
<tr>
<td>4</td>
<td>Control Structures: Part 1</td>
<td>15-18</td>
</tr>
<tr>
<td>5</td>
<td>Control Structures: Part 2</td>
<td>19-23</td>
</tr>
<tr>
<td></td>
<td>Test Two</td>
<td>24</td>
</tr>
<tr>
<td>6</td>
<td>Methods</td>
<td>25-29</td>
</tr>
<tr>
<td>7</td>
<td>Arrays</td>
<td>30-35</td>
</tr>
<tr>
<td></td>
<td>Test Three</td>
<td>36</td>
</tr>
<tr>
<td>10</td>
<td>Strings</td>
<td>37-39</td>
</tr>
<tr>
<td>14</td>
<td>Exceptions</td>
<td>40-41</td>
</tr>
</tbody>
</table>

Test Dates
Test # 1 02/11/00  Fri
Test # 2 03/08/00  Wed
Test # 3 04/14/00  Fri
Cumulative Exam 05/01/00 Mon 1:15 P.M.