| Introduction to Computer Programming in Engineering and Science | | | | | |
|---|------|-------------|---------------|--|--|
| Objectives: | 00UV | Discipline: | С | | |
| | | | Physics, Matl | | |

Introduction to Computer Drogramming in Engineering and Science

Computer Science, hematics, Biology, Chemistry **Course Code:** 360-420-DW **Ponderation:** 3-2-3 **Prerequisite:** 00UM, 00UP, 00UT **Course Credit:** 2 2/3 Corequisite: Semester: 00UQ 4

Introduction

Introduction to Computer Programming in Engineering and Science is offered to students in their final semester. The course focuses on the use of computational methods to solve complex problems in science and

Objectives and Standards for

OBJECTIVE

STANDARD

LEARNING OBJECTIVES

- 4. Correct graphical and mathematical representations of systems 5. Appropriate use of

 OBJECTIVE
 STANDARD
 LEARNING OBJECTIVES

| | OBJECTIVE | STANDARD | LEAR | NING OBJECTIVES |
|----|---------------------|----------------------------|---------|---|
| | | | 4.9.8. | Identify the order of precedence of relational and logical operators. |
| | | | 4.9.9. | Understand short circuit evaluation. |
| | | | 4.9.10. | Understand data validation and its importance to data integrity and system. |
| | | 4.10. Design a method that | 4.10.1. | Solve problems by constructing and using methods that perform one major task. |
| | | solves a single | 4.10.2. | Create a method. |
| | | identifiable task | 4.10.3. | Invoke a method. |
| | | | 4.10.4. | Pass primitive data values to a method. |
| | | | 4.10.5. | Return a single value from a method. |
| | | | 4.10.6. | Pass a reference value to a method. |
| 5. | To apply | 5.1. To solve mathematical | 5.1.1 | Use arithmetic operators respecting the rules of precedence. |
| | numerical methods | problems numerically | 5.1.1. | Convert arithmetic expressions into Java expressions. |
| | and computer | using Java | 5.1.2. | Evaluate Java arithmetic expressions. |
| | programming | | 5.1.3. | Recognize the limitations of numeric data types. |
| | techniques to solve | | 5.1.4. | Explain overflow and underflow errors. |
| | scientific and | | 5.1.5. | Cast from one primitive data type to another. |
| | engineering | | 5.1.6. | Explore and use classes from Java's standard packages. |
| | problems | | 5.1.7. | Use methods of the Math class – pow(), sqrt(), round(), random(), min(), and max(). |
| | | 5.2. Learn appropriate | 5.2.1. | Define appropriate convergence conditions for iterative methods. |
| | | numerical methods to | 5.2.2. | Understand and minimize errors related to numerical solution of mathematical equations. |
| | | computation of a model | 5.2.3. | Solve scientific models involving non-linear equations by a variety of methods. |
| | | - | 5.2.4. | Solve scientific models involving first-order ordinary differential equations (i.e. initial and |
| | | | | boundary value problems) using Euler or related methods. |
| | | | 5.2.5. | Solve other type of scientific models using the appropriate numerical techniques (e.g. by |
| | | | | Fourier or Laplace transform, Monte-Carlo, Newton-Raphson, etc.) |
| | | | 5.2.6. | Perform simple numerical optimization given a well-posed problem (e.g. by golden |
| | | | | section search method, downhill simplex method, genetic algorithm, etc.) |
| 6. | To apply a general | | | |
| | approach to model | | | |
| | development, | | | |
| | application and | | | |

| 6.2. Appropriate use of | 6.2.1. | Translate model into a form suitable for a computational solution |
|--|--------|--|
| computational methods | 6.2.2. | Identify numerical method(s) best suited to solve a given model |
| | 6.2.3. | Identify which optimization algorithm is best suited to solve a model |
| | 6.2.4. | Determine appropriate numerical parameters for solution of the model (<i>e.g.</i> sample size, time step, <i>etc.</i>) |
| | 6.2.5. | Evaluate a solution for convergence |
| 6.3. Evaluation of computational methods | 6.3.1. | Test the validity of the computational method on simpler problems that are well understood. |
| - | 6.3.2. | |
| | | |