COMP 150. OBJECT ORIENTED PROGRAMMING (4)

Four hours of lecture in the lab per week.
Prerequisite: Programming experience.
Introduction to algorithms, their representation, design, structuring, analysis and optimization. The course introduces the concept of object paradigm, design and implementation of algorithms as structured programs in a high level language.
GenEd: B4

2. Mode of Instruction.

<table>
<thead>
<tr>
<th></th>
<th>Units</th>
<th>Hours per Unit</th>
<th>Benchmark Enrollment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lecture</td>
<td>4</td>
<td>1</td>
<td>24</td>
</tr>
<tr>
<td>Seminar</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Laboratory</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Activity</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

3. Justification and Learning Objectives for the Course. (Indicate whether required or elective, and whether it meets University Writing, and/or Language requirements) [Use as much space as necessary]

The course is a required course for Computer Science majors according to accreditation guidelines.

Through this course, students will be able to

- Apply the core concepts of the object oriented programming.
- Analyze, design, implement and test programs, organized around the Central idea of the Object.
- Discuss the central idea of programming
- Use Object oriented analysis and design methodology to build models of the simple objects
- Discuss code Encapsulation as the engineering tool for ensuring code reuse, and stability..
- Participate in the programming activities, as a team member.
- Discuss their ideas on the proposed solutions of the assignments.
- Write English language comments in the source code
- Realize similarities and differences between programming and natural languages.
- Use diagrams and charts as powerful form of the the pre-language level modeling.
- Organize and express ideas clearly and convincingly in oral and written forms.

This course is not designed to satisfy the University Writing or Language requirements.

4. Is this a General Education Course  YES
If Yes, indicate GE category:
A (English Language, Communication, Critical Thinking)  
B (Mathematics & Sciences)  B4
C (Fine Arts, Literature, Languages & Cultures)
5. **Course Content in Outline Form.** *(Be as brief as possible, but use as much space as necessary)*

- Introduction to IDE, and a first program.
- Testing as a programming activity. Client use of the Object’s Api, first use of Objects.
- Variables and assignments. Data typing. Primitive data types.
- Flow of the execution. Variables, Boolean conditions and control structures. Automation of the repetitious task and selfreferencing.
- Block structure of the code. Scope of the name.
- Object Encapsulation and implementation hiding, role of the api. OO structure of the code.
- Indexed data types, arrays.
- Object view on the “smart” date structures. Lists.
- OOD: from the requirement to the api. Separation of the api and the implementation. Programming for the contract.
- Engineering benefits of OOAD.
- Subclasses. Inheritance as the refinement, and enhancement of the functionality. Thin wrappers.
- Inheritance and polymorpism of the behavior as the enrichment of the data type. Casting.
- Sorting algorithms.
- Binary searches. Recursion.

6. **References.** *(Provide 3 - 5 references on which this course is based and/or support it.)*

1) JAVA, An Intro to Computer Science and Programming by Walter Savitch : ( current edit.) Prent ceHall, ISBN 0-13-031697-0

7. **List Faculty Qualified to Teach This Course.**

   All Computer Science faculty.

8. **Frequency.**

   a. Projected semesters to be offered: Fall ___X__ Spring _X____ Summer ___X__

9. **New Resources Required.**

   a. Computer (data processing), audio visual, broadcasting needs, other equipment

      Use of existing computer lab.

   b. Library needs

      none

NEWCRSFR 9/30/02
c. Facility/space needs
   none

10. Consultation.
    Attach consultation sheet from all program areas, Library, and others (if necessary)

11. If this new course will alter any degree, credential, certificate, or minor in your program, attach a program modification.


Proposer of Course Date