Systems Programming

Academic Year: (2017 / 2018) **Review date: 25-04-2017**

Department assigned to the subject: Telematic Engineering Department

Coordinating teacher: ALARIO HOYOS, CARLOS

Type: Compulsory ECTS Credits : 6.0

Year : 1 Semester : 2

REQUIREMENTS (SUBJECTS THAT ARE ASSUMED TO BE KNOWN)

Programming (1st course, 1st semester)

OBJECTIVES

The goal of this module is for the student to deepen their knowledge in programming techniques, including advanced object orientation, as well as some of the more fundamental data structures and algorithms. The programming language the module will be based on is Java, although the acquired knowledge can be applied to other programming languages as well.

1.Design and development of applications using the Java programming
 a.Knowledge. Interpret, explain and exemplify:
  i.the elements of object orientation
  ii.the concept of recursion
  iii.the fundamentals of data structures, such as lists, stacks, queues, deques and trees
  iv.different algorithms processing data structures
 b.Specific abilities:
  i.Program object oriented applications
  ii.Represent, design and implement recursive algorithms
  iii.Implement algorithms and use linear data structures (linked lists, stacks, queues, deques, trees,etc)
  iv.Decide which data structures fit better to given problems, and reuse or design appropriate algorithms for solving specific problems
 c.Attitudes. A critical attitude towards:
  i.the suitability of different data structures and algorithms for specific problems
  ii.the use of the bibliography and other information resources for increasing his/her knowledge in other topics related to the subject
2.Use of an application environment that provides facilities for software development.
  a.Be able to create, compile and run a java program
  b.Be able to interpret and eliminate error messages and warnings shown by the compiler

DESCRIPTION OF CONTENTS: PROGRAMME

The syllabus of this module is divided into five modules:
1. Object Based programming
  1.1. Definitions: Classes and objects
  1.2. Composition of classes
  1.3. Constructors
  1.4. Shadowing
  1.5. Modifiers
2. Object Oriented Programming
  2.1. Inheritance
  2.2. Overwriting and overloading
  2.3. Casting
  2.4. Abstract classes and interfaces
  2.5. Multiple inheritance and polymorphism
3. Recursion
  3.1. Definitions
  3.2. Types (linear, non linear, mutual)
  3.3. Recursion vs. iteration
  3.4. Implementation and execution examples
4. Data structures, algorithms and implementation samples

4.1 Linked Lists and Double Linked Lists.
4.2 Stacks
4.3 Queues (simple, double and priority queues)
4.4 Tress (n-ary tress, binary trees, binary search trees)

LEARNING ACTIVITIES AND METHODOLOGY

The teaching methodology of this course is based in the active integration of the student in the teaching activity and in the application of the principles of continuous evaluation.
The course is divided into units, where each unit consists of lectures in large groups and lab sessions in small groups. All the necessary information and material will be available from the web site of the course. The student participation in the class activities (questions, problem-solving in groups) will be positively taken into account. The lab scripts with the exercises and programs to be carried in the lab are also available in advance. Additionally to the lectures and lab sessions, the web page proposes some homework activities.
1. Lecture sessions (theory, exercises and questions). In these classes the students will be introduced to the educational contents they should learn. In order to facilitate the understanding, the students will be provided with the class notes and basic reference books.
2. Lab sessions (Problem solving). In these classes in computer labs, the students will analyze and develop programs applying the different concepts taught in the lectures
3. Personal work: Self-study to understand the theoretical concepts and how to apply them, homework, to solve exercises
4. Exams: Exercises to be solved by the students to demonstrate the knowledge and the required abilities acquired during the course.
a) Mid-term exams
b) Final exam
5. Tutoring (PO:a) according to current regulations

ASSESSMENT SYSTEM

The evaluation of the course is based on midterm and final exams.
The distribution of the marking is based on the following criteria:
1. 50%: continuous evaluation. Mid-term exams to evaluate overall knowledge (theory and problems for several units.
2. 50% Final exam: Overall knowledge of the student for the whole course
(*) Attending final exam is mandatory and the average with coninuos grades only will be calculated when the grade obtained in the final exam is higher than 3.5/10

In any case, current assessment legislation adopted on May 31, 2011 will be applied
http://www.uc3m.es/portal/page/portal/organizacion/secret_general
ormativa/estudiantes/estudios_grado
ormativa-evaluacion-continua-31-05-11_FINALx.pdf

During the evaluation process, all the students must follow those standards of conduct that are consistent with the ethical values of the university reflected in these documents:
http://www.uc3m.es/portal/page/portal/conocenos
uestros_estudios/grados/tu_compromiso_universidad .
http://www.uc3m.es/portal/page/portal/conocenos/guia_buenas_practicas

Specifically, the student is expected to respect the course evaluation rules and reject any fraudulent behaviour, such as any means of copying in the assessment tests or plagiarizing papers or assignments. Likewise, the student is responsible for guarding their personal work to prevent this fraudulent behavior by peers.

Any behaviour that fails to comply with these guidelines will be penalized.
http://www.uc3m.es/portal/page/portal/organizacion/secret_general
ormativa/estudiantes/estudios_grado/Acuerdo%20situaciones%20defraudaci%F3n%20ex%E1menes-2002.pdf

% end-of-term-examination:	50
% of continuous assessment (assigments, laboratory, practicals…):	50

BASIC BIBLIOGRAPHY

- Mark A. Weiss Data Structures and Problem Solving Using Java, Addison-Wesley.

- Michael T. Goodrich and Roberto Tamassia Data Structures and Algorithms in Java, John Wiley & Sons, Inc..

BASIC ELECTRONIC RESOURCES

- Carlos Delgado Kloos, Carlos Alario Hoyos, Iria Estévez Ayres, Carmen Fernández Panadero, Julio Villena Román... . Introduction to Programming with Java (MOOCs en edX): https://www.edx.org/xseries/introduction-programming-java