Subject:	Object-Oriented Programming	**Code**	72.33
Credits:	6		
Department	Digital Systems and Data	**Version**	2019

Course: Computer Science Engineering, Mechanical Engineering

Curriculum: M22, S10 A - Rev18, S10-Rev23, S10 - Rev18

Objectives:

No.	Description
1	Distinguish the advantage of using object-oriented programming in design and implementation of programs versus other paradigms.
2	Design and implement object hierarchies that are an abstraction of the real world and allow for solving problems.

Contents:

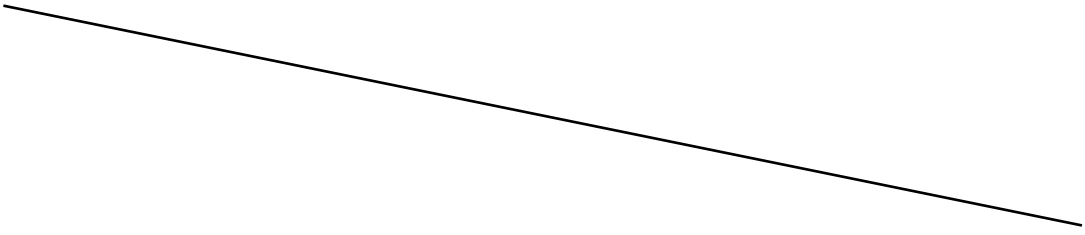
Abstract Data Type: information hiding and encapsulation. Characteristics of the Object Paradigm: encapsulation, hiding, inheritance and polymorphism. Messages and methods. Classes, subclasses and instances. Internal representation of objects and classes. Type checking. Static and dynamic binding. Use and construction of libraries. Collections (arrays) and iterators. Handling of exceptions. Object persistence. Introduction to UML for class and interface design. Comparison of programming paradigms. Application workshop with an object-oriented language such as Smalltalk or Java.

Required bibliography:

No.	Description
1	No bibliography has been uploaded.
2	Ruby Tutorial: https://www.tutorialspoint.com/ruby/index.htm

Optional bibliography:

No.	Description
1	No optional bibliography has been uploaded.
2	David Flanagan & Yukihiro Matsumoto. The Ruby Programming Language, O'Reilly, USA 2008
3	The Ruby style guide: https://github.com/bbatsov/ruby-style-guide



Subject:	Object-Oriented Programming	**Code**	72.33
Credits:	6		
Department	Digital Systems and Data	**Version**	2019

Course transcript:

No.	Description
1	**Introduction to OOP** Background of the object paradigm: abstract data type. Characteristics of object-oriented programming (OOP): (OOP): encapsulation, information hiding, polymorphism and inheritance. Object-oriented paradigm solution to the crisis: reusability.
2	**OOP terminology** Classes (casts), objects (instances) and messages (communication between objects). Class and instance variables and methods. Superclasses and subclasses. Abstract classes. Composition and inheritance. Message overloading and overwriting. Dynamic vs. static binding. Object persistence. Basic UML concepts
3	**Introduction to Ruby** Development, compilation and execution environment. Operators. Decision and control structures. Collections.
4	**Introduction to Java** General characteristics of Java as a hybrid object-oriented language. Concept of virtual machine: bytecode.
5	**Fundamental Elements of the Java Language** Primitive data types. Operators. Expressions. Operator precedence. Assignment. Program flow control: decision, repetition and jumps. Classes and objects in Java. Objects lifetime and scope. Constructors. Class and instance variables and methods. Method overloading. Object finalization and Garbage Collection. Construction of a Java program. Array handling. Examples of existing classes: System and String. Violation of the OOP paradigm: operators and statements for flow control are not. messages
6	**Information Hiding** Package concept. Access specifications: public, private and protected. Nested classes. Interfaces: their implementation. Violation of the OOP paradigm: access to public instance variables from outside the class.
7	**Inheritance** Inheritance syntax in Java. Order of constructors invocation. The final modifier. Methods over-writing vs. vs. Methods overloading. Abstract classes. The Object class. Existing class hierarchies in Java. Java. Examples of classes: Random, Stack, Vector and Hashtable.
8	**Exceptions** Error handling. Basic exceptions. Exception catching and handling in Java. Construction of safe programs. **Input/Output System**

9	Review of the data flow concept: stream. Classes for streams handling. Object persistence.
10	**Unit testing** Test Driven Design. JUnit.
11	**User interface.** Graphical interface. Introduction to JavaFX. Concept of event. Listeners.

Practical assignments:

Subject:	Object-Oriented Programming	**Code**	72.33
Credits:	6		
Department	Digital Systems and Data	**Version**	2019

No.	Description
1	**Practical assignments:** PA No. 01: Introduction to Ruby PA No. 02: Ruby Implementations: PA No. 03: Introduction to Java PA No. 04: Classes, Objects and Methods in Java PA No. 05: Inheritance and Polymorphism in Java PA No. 06: Exceptions and Interfaces in Java PA No. 07: Generics and Collections in Java PA No. 08: Advanced Java Design

Laboratory assignments:

No laboratory assignments.

Professor in charge:	Garberoglio, Marcelo Fabio
Head of Department:	Bolo, Mario Enrique

CERTIFIED to be a true translation into English of the copy of the documents in Spanish I had before me in the city of Buenos Aires, on the 23th. day of December, 2024. ***Armando Javier Forte - Sworn Translator. CTPCBA Registration No. 3553.*** ----------

ES TRADUCCIÓN FIEL al idioma inglés de la copia de los documentos en idioma español que he tenido ante mí y a los cuales me remito, en la ciudad de Buenos Aires, a los 23 días del mes de diciembre de 2024. ***Armando Javier Forte - Traductor Público. CTPCBA Inscripción: 3553.***----------