Course unit title:	The JAVA programming language
Course unit code:	CSC342
Type of course unit: **(Compulsory/optional)**	Compulsory
Level of course unit **(First, second or third cycle)**	Bachelor (1st cycle)
Year of study:	3
Semester when the unit is delivered:	5 or 6 or 7
Number of ECTS credits allocated:	6
Name of lecturer(s):	TBA

Learning outcomes of the course unit:

Upon successful completion of this course students should be able to:

- Recall details of the concept of platform independent software, the Java Virtual Machine and intermediate byte code
- Apply the syntax of the language and compare it to other typical procedural (like C and Pascal) and object-oriented (like C++) languages.
- Develop medium-size desktop applications and persistent storage (text files and binary files) in the back-end.
- Use advanced features of the language such as threads and networking in the development of small programs.
- Use Java for programming different platforms such as internet of things.

Mode of delivery:	Face-to-face
Prerequisites and co-requisites:	CSC205 or CSW205
Recommended optional program components:	None

Course contents:

Objective:
This course examines Java as a general purpose programming language, The course gives emphasis on Java's advanced data structures design.

Description:

The Scope of Java:
The Internet and the Web, Internet of Things, smartphones. Comparison of Java with other programming languages. Object-oriented programming review. A first Java program.

Java Simple Types, Variables, and Statements:
Java primitive types, tokens, operators, expressions; Java simple and control statements; Java strings and arrays.

Java objects and classes:
Abstract data types and data structures, advanced classes in Java.

Java Object-Oriented Programming:
Extending Java classes, inheritance, overloading, overriding, polymorphism.

Programming in the large - packages:
Introduction to Packages. Packages and Files. Using and Identifying Packages. Packages Scope Rules.

Exception handling in Java:
Introduction. Exceptions and Objects. Exceptions and Classes.

Threads - concurrent programming with Java:
Introduction. Multithreading. Independent threads. Scheduling, thread priorities. Thread Interaction. Interruptions. Deadlock.

Applications using JAVA:
Graphical User Interfaces. Graphics, Animation Tools, and Sound. User interface, Internet of Things, Arduino, raspberry Pi, android.

Recommended or required reading:	Deitel P., "Java How to Program: Late Objects Version", Prentice Hall, 2010 Ullman L., "Modern JavaScript: Develop and Design", Peachpit Press, 2012 Chin S., Weaver J., "Raspberry Pi with Java: Programming the Internet of Things (IoT)", McGraw-Hill Education, 2015. Lewis J., Loftus W., "Java software solutions, Foundation of program design", Pearson, 2007 Koffman E., Wolz U., "Problem solving with Java", Addison-Wesley Barnes D.J., Kőlling M., "Objects First with Java, a Practical introduction using BlueJ", Pearson Education, 2006 Eckel B., "Thinking in Java", Pearson Education, 2006 Reed Doke E., Satzinger J.W., Rebstock Williams S., "Object-Oriented Application Development using Java", Course Technology, 2002

Planned learning activities and teaching methods:	Class Instruction: Consultation:	42 Hours 15 Hours
Assessment methods and criteria:	Examinations Class Participation / Assignments	70% 30% 100%
Language of instruction:	English	
Work placement(s):	No	
Place of Teaching:	Regular Classroom European University Cyprus, Nicosia Computer Laboratory European University Cyprus, Nicosia	