THE UNIVERSITY OF BUCKINGHAM

Sarajevo School of Science and Technology

MODULE SPECIFICATION

Name of Module	Object-Oriented Programming (OOP)						
Parent School/Dept	Computer Science/Information Systems						

Programme(s) where module is offered	BSc Computer Science with Electrical Engineering;
	BSc Computer Science with Economics;
	BSc Computer Science with Business;
	BSc Computer Science with International Relations;
	BSc Computer Science with Political Science;
	BSc Information Systems with Electrical Engineering;
	BSc Information Systems with Economics;
	BSc Information Systems with Business;
	BSc Information Systems with International Relations;
	BSc Information Systems with Political Science;						

Status (core, option, free choice)	Core			**Pre-Requisite Modules or Qualifications**		CSIS 110 CSIS 160	
FHEQ Level	4	**Unit Value**	6 ECTS	**Module Code**	**CS280**	**Module coordinator**	Dr. Elmedin Selmanović
Term taught	Fall or Spring			**Applicable From**		2016	

Educational Aims of the Module

This module teaches the art of object-oriented programming (OOP) using Java language. As CS/IS 110 and 160 are prerequisite for this module, students who attend this class will have knowledge of programming and problem solving in C++. In the first few weeks, the module is going to concentrate on reinforcing the basics of object-oriented programming in Java. For the rest of the module, the students are going to learn advanced Java programming concepts that will enable them to solve software engineering problems. Some of these advanced topics will be study of generic collections, streams, regular expressions, event-driven programming, and others. Laboratory work and programming assignments are an integral part of this module.

Module Outline/Syllabus

- Brief introduction to Java and its syntax
- Classes and objects
- Extending classes with inheritance
- Abstract classes and interfaces
- Polymorphism
- Exceptions
- Graphics and event-driven programs
- Regular expressions
- Files, streams, and object serialization
- Generic collections
- Generic classes and methods (templates)

Student Engagement Hours

Type	Number per Term	Duration	Total Time
Lectures	30	2 hours	60
Laboratory sessions	15	2 hours	30
		Independent Learning Hours	**60**
		Total Contact Hours	**90**
		Total Engagement Hours	**150**

Assessment Method Summary

Type	Number Required	Duration / Length	Weighting	Timing/Submission Deadline
Final exam	1	180 minutes	50%	End of semester
Mid-term exam	1	60 minutes	20%	Mid-semester
Assignment	10	60 minutes	20%	Weeks 1 through 15
Test	2	60 minutes	10%	Week 5 and 12

Module Outcomes		
Intended Learning Outcomes: 1. Theoretical knowledge of OOP programming concepts in Java 2. Intelligently use and discuss standard coding conventions --- indentation, naming conventions, program and application structure, etc. 3. Make intelligent programming and system design decisions	→	**Teaching and Learning Strategy:** 1. Lectures on module material (ILO 1-3) 2. Research/investigation assignments for self-study engagement 3. Lab exercises with programming tools and practice problems (ILO 1-3) 4. Group project assignment (ILO 1-3)
	→	**Assessment Strategy** 1. Mid-term exam (ILO:1-2) 2. Final exam (ILO: 1-3) 3. Assignment (ILO: 1-3) 4. Test (ILO: 1-3)
Practical Skills 1. Effectively read and analyse Java programs and applications 2. Program rapidly and effectively in Java using OOP programming concepts 3. Interpret and demonstrate understanding of computer program writing and testing methodologies	→	**Teaching and Learning Strategy:** 1. Laboratory sessions with tutor-lead support (PS 1-3) 2. Assignment (PS 1-3) 3. Test (PS 1-3)
	→	**Assessment Strategy** 1. Mid-term exam (PS: 1-3) 2. Assignment (PS 1-3) 3. Test (PS 1-3) 4. Final exam (PS:1-3)
Transferable Skills 1. IT skills 2. Problem solving using OOP 3. Presentation skills 4. Ability to work as part of a team	→	**Teaching and Learning Strategy:** 1. Lab exercises (TS: 1,2,4) 2. Lectures (TS:1-4) 3. Assignment (TS: 1,2,4)
	→	**Assessment Strategy** 1. Assignment (TS:1-4) 2. Test (TS:1, 2)

Key Texts and/or other learning materials
Set Text • Deitel, Deitel,(2015) “Java - How to Program,” 10th Edition, Prentice Hall, **Supplementary Materials** • Bloch, J., (2008) “Effective Java,” Second Edition, Addison-Wesley. • Schildt,H., (2011). “Java, A Beginner's Guide,” Fifth Edition, McGraw-Hill, 2011

Please note: This specification provides a concise summary of the main features of the module and the learning outcomes that a typical student might reasonably be expected to achieve and demonstrate if he/she takes full advantage of the learning opportunities that are provided. More detailed information on the learning outcomes, content and teaching, learning and assessment methods of each module and programme can be found in the departmental or programme handbook. The accuracy of the information contained in this document is reviewed annually by the University of Buckingham and may be checked by the Quality Assurance Agency.

Date of Production	Autumn 2016
Date approved by School Learning and Teaching Committee	28th September 2016
Date approved by School Board of Study	12th October 2016
Date approved by University Learning and Teaching Committee	2nd November 2016
Date of Annual Review	December 2017