|  |  |  |
| --- | --- | --- |
| dau_logo_BW | **EASTERN MEDITERRANEAN UNIVERSITY**  **SCHOOL OF COMPUTING AND TECHNOLOGY**  **DEPARTMENT OF INFORMATION TECHNOLOGY**  **COURSE POLICY SHEET** |  |

|  |  |
| --- | --- |
| **Course Title** | Multi Platform Programming |
| **Course Code** | ITEC314 |
| **Type** | Full Time |
| **Semester** | Fall/Spring |
| **Category** | AC (Area Core) |
| **Workload** | 210 Hours |
| **EMU Credit** | (3,2,0) 4 |
| **Prerequisite** | ITEC243 |
| **Language** | English |
| **Level** | Third Year |
| **Teaching Format** | 3 Hours Lecture and 2 Hour Lab per week |
| **ECTS Credit** | 7 |
| **Course Web Sites** | <http://courses.sct.emu.edu.tr/itec314> and  [http://staff.emu.edu.tr/cemyagli/en/teaching/itec314](https://staff.emu.edu.tr/cemyagli/en/teaching/itec314) |

|  |  |  |  |
| --- | --- | --- | --- |
| **Instructor(s)** | Cem Yağlı | **Office Tel** | +90 392 6301137 |
| **E-mail** | Cem.yagli @emu.edu.tr | **Office No** | CT109 |

|  |
| --- |
| **Course Description** |
| This course is aiming to introduce students to multi-platform (cross platform) application development, including the reasons of that study, the approaches and techniques for meeting the requirements. The fundamentals and alternative ways of the multi-platform programming with restrictions and benefits are also taught in the course. The given theory is supported with exercises and sample applications using Java programming language (J2SE) that is the most popular alternative solution of today. Students get experience on “Write once and run everywhere” approach of programming. |

|  |
| --- |
| **General Learning Outcomes** |
| On successful completion of this course students should be able to:   * Explain the most common problems of software developers coding for a unique solution aiming to work on different operating systems. * Explain the multiplatform (cross platform) programming with its requirements, restrictions and benefits. * Explain the alternative approaches, methods and techniques for solving multiplatform programming problems of today. * Describe the concept of “virtual machines”, how they are working, configuring and maintaining. * Explain why the Java programming language is the most popular alternative solution of today for the multiplatform programming problem. * Analyse, design and implement a desktop application (using J2SE) that can be work on different operating systems. * Code in Java programming language (J2SE) to develop a software (SW) solution for a multiplatform. * Apply the structured and object-oriented with event-driven programming skills to SW development projects. |

|  |
| --- |
| **Teaching Methodology / Classroom Procedures** |
| * Each week there are three lecture hours, and two lab hours. * Laboratory works are organized to go as parallel with the theory, given in lecture hours in the classroom. * Students’ performance evaluated by:   + - Midterm-1 covers chapters 1, 2, 3, 4, 5 and 6. (30%)     - Midterm-2 covers chapters 7, 8 and 9 (30%)     - Final covers 10, 11, 12, 13, 14 and 15 (40%) * All course related material (reading texts, tutorials, previously asked exam questions with their solutions, and announcements can be reached by students through the course WEB site (http://courses.sct.emu.edu.tr/itec314 ). |

|  |
| --- |
| **Course Materials / Main References** |
| ***Text Book:***  Malhotra A., Choudhary S., “Programming in Java”, Oxford University Press, Second Edition (2014),  ISBN: 9780198094852  ***Resource Books (Available in EMU Library, Reference Section):***   1. C.Thomas Wu, “An Introduction to Object-Oriented Programming with JAVA”, McGraw-Hill International Edition, Fifth Edition (2010), ISBN: 9780073523309 2. Bruce Eckel, “Thinking in Java”, Fourth Edition, Prentice Hall, 2006, ISBN: 9780131872486 3. Evans B. J., Flanagan D., “Java In A Nutshell”, Sixth Edition, O'Reilly Media, 2014. ISBN: 9781449370824 4. David Flanagan, “Java Examples in a Nutshell”, Third Edition, O'Reilly Media, 2004. ISBN: 978-0596006204 5. Robert F. Stärk , Joachim Schmid, Egon Börger , "Java and the Java Virtual Machine: Definition, Verification, Validation", Springer,2001, ISBN: 9783642594953   ***Lecture Notes:***  All lecture materials are also available online in Adobe PDF (Portable Document Format). |

|  |  |
| --- | --- |
| **Weekly Schedule / Summary of Topics** | |
| **1 week** | **.The need for multiplatform programming:** Operating System (OS) – Application platforms (multiplatform). The “Write once run anywhere” though. The alternative approaches and solutions for multiplatform problem. Virtual machines. How Java Virtual Machines are working. Java editions. Java programming language essentials.  **.The programming languages in time:** Historical changes through the Structured Programming to the Object Oriented Programming language.  **Object Oriented Programming in Java:** Classes and objects. Messages and methods. Class and instance Data values. Inheritance. |
| **1 week** | **Introduction to Java:** Obtaining and installing Java virtual machine and NetBeans IDE. The components of a Java program. The syntax. Edit-Compile-Run cycle. Java standard classes.  **The basic input-output methods in Java:** console I/O methods. Dialog box I/O methods.  **Numerical data in Java:** Variables. Arithmetic expressions. Constants. Getting numeric input values. Standard output. Standard input. The Math Class. Random number generator. GregorianCalendar and Date classes. |
| **1 week** | **User defined Classes in Java:** Defining and using a user defined class. Multiple classes. Matching arguments and parameters. Passing objects to a method. Constructors. Information hiding and visibility modifiers (public/private). Class constants. Local variables. Calling methods of the same class. Changing any class to main class. Returning and object from a method. The reserved word “this”. Overloaded methods and constructors. Class variables and methods. Call-by-value parameter passing. Organizing classes into a package. Using JavaDoc comments for class documentation.  **Inheritance and Polymorphism:** Defining classes with inheritance. Using classes effectively with polymorphism. Inheritance and member accessibility. Inheritance and constructors. Abstract super classes and abstract methods. Inheritance versus interface. |
| **2 weeks** | **Fundamentals of Coding-I:**  **.Selection Statements:** The if statement. Nested if statements. Boolean expressions and variables. Comparing objects. The switch statement.  **.Repetition Statements**: The while statement. Pitfalls in writing repetition statements. The do-while statement. Loop and a half repetition control. Confirmation dialog. The for statement. Nested for statements. |
| **2 weeks** | **Fundamentals of Coding-II:**  **. Nested selection and looping**  **. Logical (Boolean) variables of Java and Logical operators and operations**  **.Formatted Output** |
| **2 weeks** | **Midterm Exams** |
| **2 week** | **.Exception and Assertions:** Catch exceptions. Throwing exceptions and multiple catch blocks. Propagating exceptions. Types of exceptions. Programmer defined exceptions. Assertions.  **.Collections in Java**  **.Recursive Algorithms:** Basic elements of recursion. Advantages of recursion. When not to use recursion. |
| **1 week** | **Characters and Strings:** Characters. Strings. Pattern matching and regular expressions. The Pattern and Matcher classes. Comparing strings. StringBuffer and StringBuilder classes.  **String manipulation**: Algorithms and methods. |
| **1 week** | **Arrays:** Array basics. Arrays of objects. Passing arrays to methods. Two-dimensional arrays. Lists and Maps.  **Basic search and sort** Algorithms and techniques with arrays. |
| **1 week** | **File Input and Output:** File and JFileChooser Objects. Low level File I/O. High level File I/O. Text Files I/O. Object Files I/O. |
| **1 week** | **Visual Design and Event Driven programming:** TheGUI elements, Form design and handling the events. |
| **3 weeks** | **Final Exams** |

|  |
| --- |
| **Requirements** |
| * Each student can have only one make-up exam. * One who misses an exam should provide a medical report or a valid excuse within 3 days after the missed exam. * The make-up exam is done at the end of the term and covers all the topics. * Students who fail to attend the lectures regularly may be given NG grade. * Once the grades are announced, the students have only one week to do objection about their grades. * It is the students’ responsibility to follow the announcement in the course web site. * Exam scores are announced at the WEB site <http://students.emu.edu.tr>. * Students have to check their exam papers until the end of the week following the announcement of exam scores. |

|  |  |  |  |
| --- | --- | --- | --- |
| **Method of Assessment** | | | |
| **Evaluation and Grading** | **1st Midterm Exam** | **2nd Midterm Exam** | **Final Exam** |
| **Percentage** | 30 % | 30 % | 40 % |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Grading Criteria \*** | | | | | | | | | | | |
| **A** | **A-** | **B+** | **B** | **B-** | **C+** | **C** | **C-** | **D+** | **D** | **D-** | **F** |
| 90 -100 | 85 - 89 | 80 - 84 | 75 - 79 | 70 - 74 | 65 - 69 | 60 - 64 | 56 - 59 | 53 - 55 | 50 - 52 | 40 - 49 | 0 – 39 |

\* Letter grades will be decided upon after calculating the averages at the end of the semester and distribution of the averages will play a significant role in the evaluation of the letter grades.