| COURSE INFORMATON | | | | | |
|---|---|---|---|---|---|
| **Course Title** | *Code* | *Semester* | *L+P Hour* | *Credits* | *ECTS* |
| **THEORY OF ALGORITHMS** | **CSE511** | | **3+0** | **3** | **10** |

| | |
|---|---|
| **Prerequisites** | |

| | |
|---|---|
| **Language of Instruction** | English |
| **Course Level** | Master's Degree |
| **Course Type** | Technical Elective |
| **Course Coordinator** | |
| **Instructors** | Assist. Prof. Onur Demir |
| **Assistants** | |
| **Goals** | The aim of this course is to provide students with knowledge to introduce well-known algorithmic design techniques and well-known algorithms, to analyze sequential and recursive algorithms, to utilize well-known algorithms for similar problems, and to introduce theory of NP. |
| **Content** | Concepts of designing algorithms and complexity analysis of algorithms, solving recurrence equations and formal proofs, an intuitive and formal introduction to the concept of order and growth, brute force approach, divide and conquer approach, dynamic programming, greedy approach, graph algorithms, theory of NP. |

| **Course Learning Outcomes** | **Program Learning Outcomes** | **Teaching Methods** | **Assessment Methods** |
|---|---|---|---|
| 1) Adequate knowledge in analyzing algorithms; ability to use theoretical and empirical methods to analyze both sequential and recursive algorithms; Adequate knowledge in theory of NP. | 1,4,5 | 1,2 | A,C |
| 2) Adequate knowledge in algorithmic design techniques, algorithmic solutions to basic problems; | 1,4,5 | 1,2 | A,C |
| 3) Ability to use design techniques to model and solve problems; Ability to apply basic algorithms to more complex problems | 4,5 | 1,2 | A,C,D |
| 4) Ability to devise, select, and use modern techniques and tools needed for the design and implementation of algorithms. | 5,11 | 1,2 | A,C,D |
| 5) Ability to analyze scientific publications. | 4 | 4 | D |

| | |
|---|---|
| **Teaching Methods:** | 1: Lecture, 2: Question-Answer, 3: Lab, 4: Case-study |
| **Assessment Methods:** | A: Testing, B: Experiment, C: Homework, D: Project |

| COURSE CONTENT | | |
|---|---|---|
| **Week** | **Topics** | **Study Materials** |
| 1 | THEORETICAL BACKGROUND | Textbook, Lecture Notes |
| 2 | EFFICIENCY, ANALYSIS AND ORDER | Textbook, Lecture Notes |
| 3 | RECURRENCE | Textbook, Lecture Notes |
| 4 | RECURRENCE II | Textbook, Lecture Notes |
| 5 | BRUTE FORCE ALGORITHMS | Textbook, Lecture Notes |
| 6 | DIVIDE AND CONQUER I | Textbook, Lecture Notes |
| 7 | DIVIDE AND CONQUER II, MIDTERM I | Textbook, Lecture Notes |
| 8 | DYNAMIC PROGRAMMING I | Textbook, Lecture Notes |
| 9 | DYNAMIC PROGRAMMING II | Textbook, Lecture Notes |
| 10 | GREEDY APPROACH | Textbook, Lecture Notes |
| 11 | GRAPH ALGORITHMS I, MIDTERM II | Textbook, Lecture Notes |
| 12 | GRAPH ALGORITHMS II | Textbook, Lecture Notes |
| 13 | THEORY OF NP | Textbook, Lecture Notes |
| 14 | REVIEW | Textbook, Lecture Notes |

| RECOMMENDED SOURCES | |
|---|---|
| **Textbook** | R. Neapolitan, and K. Naimipour, Foundations of Algorithms |
| **Additional Resources** | Lecture Notes: http://cse.yeditepe.edu.tr/v2/en/academic/course-pages |

| MATERIAL SHARING | |
|---|---|
| **Documents** | |
| **Assignments** | |
| **Exams** | |

| ASSESSMENT | | |
|---|---|---|
| **IN-TERM STUDIES** | **NUMBER** | **PERCENTAGE** |
| Mid-terms | 2 | 61 |
| Programming Assignment | 3 | 22 |
| Homework | 5 | 7 |
| Presentation | 1 | 10 |
| **Total** | | **100** |
| **CONTRIBUTION OF FINAL EXAMINATION TO OVERALL GRADE** | | 30 |
| **CONTRIBUTION OF IN-TERM STUDIES TO OVERALL GRADE** | | 70 |
| **Total** | | **100** |

| | |
|---|---|
| **COURSE CATEGORY** | Expertise/Field Courses |

| | COURSE'S CONTRIBUTION TO PROGRAM | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| No | Program Learning Outcomes | Contribution | | | | | | |
| | | 0 | 1 | 2 | 3 | 4 | 5 |
| 1 | Ability to reach wide and deep knowledge through scientific research in the field of Computer Science and Engineering, evaluate, interpret and apply. | | | | | | **X** |
| 2 | Ability to use scientific methods to cover and apply limited or missing knowledge, and to integrate the knowledge of different disciplines. | | | | | | |
| 3 | Ability to construct Computer Science and Engineering problems, develop methods to solve the problems and use innovative methods in the solution. | | | | | | **X** |
| 4 | Ability to develop new and/or original ideas and algorithm; develop innovative solutions in the design of system, component or process. | | | **X** | | | |
| 5 | Ability to have extensive knowledge about current techniques and methods applied in Computer Engineering and their constraints. | | | | | | **X** |
| 6 | Ability to design and implement analytical modeling and experimental research, solve and interpret complex situations encountered in the process. | | | | | | |
| 7 | Ability to use a foreign language (English) at least at the level of European Language Portfolio B2 in verbal and written communication. | | | | | | |
| 8 | Ability to lead in multidisciplinary teams, develop solutions to complex situations and take responsibility. | | | | | | |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 9 | Ability to pass process and the results in Computer Science and Engineering field, in national and international area in or outside of the field, systematically and clearly in written or oral form. | | | | | | |
| 10 | Awareness of the social, legal, ethical and moral values, and the ability to conduct research and implementation work within the framework of these values. | | | | | | |
| 11 | Awareness of the new and emerging applications in Computer Science and Engineering field, and the ability to examine them and learn if necessary. | | | **X** | | | |
| 12 | Ability to describe the social and environmental dimensions of Computer Science and Engineering applications. | | | | | | |

| **ECTS ALLOCATED BASED ON STUDENT WORKLOAD BY THE COURSE DESCRIPTION** | | | |
|---|---|---|---|
| Activities | Quantity | Duration (Hour) | Total Workload (Hour) |
| Course Duration (Excluding the exam weeks: 12x Total course hours) | 12 | 3 | 36 |
| Hours for off-the-classroom study (Pre-study, practice) | 14 | 5 | 70 |
| Midterm examination | 2 | 3 | 6 |
| Homework | 5 | 6 | 30 |
| Programming Assignment | 3 | 25 | 75 |
| Presentation | 1 | 20 | 20 |
| Final examination | 1 | 3 | 3 |
| **Total Work Load** | | | 240 |
| **Total Work Load / 25 (h)** | | | 9.6 |
| **ECTS Credit of the Course** | | | 10 |