Information credits credits Department Physics and electronics Program Academic semester 2022 spring Lecturer Associate Prof Gulyev Mazahim E-mail: mazahim.culycv@gmail.com Phone number: 4-994 55567 70 74 Lecture room/Schedule 11 Mebseti Street, X21096 Baku, Azerbaijan (Neffchilar campus), noom Course language English Type of the subject Major Studiotal Textbooks and additional 1. C++ Programming: Program Design Including Data Structures, 6th Ed. D.S. Malik, Course Textbooks and tockicon to Programming with C++. 3vt Edition by Y. Daniel Liang, Pearson, 2014 4. C: The Complete Reference, Herbert Schildt, McGrawHill, 2016 Teaching Lecture Mathematical of Computer programming with C++. 3vt Edition by Y. Daniel Liang, Pearson, 2014 4. C: The Complete Reference, Herbert Schildt, McGrawHill, 2016 X Group discussions at seminars X Assessment Components Date/ Deadline Percent (%) Attendance 5 Ouizzes During the semster 20 Outcare participation<	General	Title and code of subject, number o	f EENG245 Fundamentals of G	EENG245 Fundamentals of Computer Programming 4		
Department Program Bachelor Academic senester 2022 spring Academic senester 2022 spring Lecturer Assosiate Prof Gultyev Mazahim E-mail: mazahim guliyev@gmail.com Phone number: +994 55567 70 74 Lecturer com/Schedule 11 Mchesti Street, AZ1096 Baku, Azerbaijan (Neftchilar campus), room Course language English Type of the subject Major aubject Tectbooks and rechnology, 2011 1. C++ Programming: Program Design Including Data Structures, 6th Ed. D.S. Malik, Course Technology, 2011 2. Fundamentals of Computer programming with C++. Settin Nakov 2013 3. Introduction to Programming with C++. Settin Nakov 2013 3. Introduction to Programming with C++. Settin Nakov 2013 4. C: The Complete Reference, Herbert Schildt, McGrawHill, 2016 Teaching Active participation At each lesson Active participation At each lesson 5 Quizzes During the semester 20 Attendance 5 Midterm exam 30 Final Tios course introduces students to the	information	credits	credits	credits		
Program Bachelor Academic semester 2022 spring Lecturer Associate Prof Guliyev Mazahim E-mail: mazahin.guliyev@gmail.com Phone number: +994.55567 70.74 Lecture room/Schedule 11 Mehseti Street, AZ1096 Baku, Azerbaijan (Neftchilar campus); room Course language English Type of the while Major subject Textbooks: Texthooks and additional materials Textbooks: 1. C++ Programming: Program Design Including Data Structures, 6th Ed. D.S. Malik, Course Texthoolegy, 2011 2. Fundamentals of Computer programming with C++. Svettlin Nakov 2013 3. Introduction to Programming with C++. 3rd Edition by Y. Daniel Liang, Pearson, 2014 4. C: The Complete Reference, Herbert Schildt, McGrawHill, 2016 Xacement X methods Group discussions at seminars X Assessment Active participation At cach lesson 5 Quizzes During the semester 20 Attendance 100 100 Final This course introduces students to the field of computer science and an eq		Department	Physics and electronics			
Academic senseter 20/2 spring Iccturer Assosiate Prof Guliyev Mazahim E-mail: mazahim.guliyev@gmail.com Phone number: 4994 55567 70 74 Iccture room/Schedule 11 Melseti Street, A21096 Baku, Azerbaijan (Neftchilar campus), room Course language English Type of the subject Major additional 1. C++ Programming: Program Design Including Data Structures, 6th Ed. D.S. Malik, Course Technology, 2011 Textbooks and additional 1. C++ Programming: With C++, Svetlin Nakov 2013 3. Introduction to Programming with C++, Svetlin Nakov 2013 3. Introduction to Programming with C++, 3rd Edition by Y. Daniel Liang.,Pearson,2014 4. C. The Complete Reference, Herbert Schildt, McGrawHill, 2016 X methods Group discussions at seminars X Assessment Components Date/ Deadline Percent (%) Attendance During the senseter 20 Midterm exam 30 Final exam 30 Final exam 100 Final exam 400 Course drive programming projects. The course will culminate in a comprehesity vill culminate in a comprehesity vill culminate in a comprehesit		Program	Bachelor	Bachelor		
Lecturer Assositate Prof Cullyev Mazahim E-mail: mazahim cullyev@genali.com Phone number: 4994 55567 70 74 Lecture room/Schedule 11 Mehseti Street, AZ1096 Baku, Azerbaijan (Neftchilar campus), room Course language English Type of the subject Major subject Textbooks and additional I. C++ Programming: Program Design Including Data Structures, 6th Ed. D.S. Malik, Course Technology, 2011 2. 2. Fundamentals of Computer programming with C++. Svetlin Nakov 2013 3. 3. Introduction to Programming with C++. Svetlin Nakov 2014 4. 4. C: The Complete Reference, Herbert Schildt, McGrawHill, 2016 X Teaching Kertwe participation At each lesson Matter azan At each lesson 5 Quizzes During the semester 20 Attendance 5 Milter exan 30 Final exam 40 5 Outgres along within computent science such as anetworks, Al, robotics, project within computent science such as networks, Al, robotics project of compute architecture will be integrated throughout the course. At integrates the concepts stuppt in the ecourse to fundiscins and embedde systems will be covered through relevant program		Academic semester	2022 spring	2022 spring		
P-mail: Imazhim gulyzv&gmail.com Phone number: 4994 55567 70 74 Lecture room/Schedule 11 Mehseti Street, AZ1096 Baku, Azerbaijan (Neftchilar campus), room Course language English Type of the subject Textbooks and additional Incommentation of the subject Textbooks and additional Incommentation of the subject Textbooks and additional sof Computer programming with C++. Svetlin Nakov 2013 3. Introduction to Programming with C++. Svetlin Nakov 2013 3. Introduction to Programming with C++. Svetlin Nakov 2013 4. C: The Complete Reference, Herbert Schildt, McGrawHill, 2016 Teaching methods Group discussions at seminars X Active participation At cach lesson 5 Quizzes During the senserer 20 Attendance 100 This course introduces students to the field of computer science and engineering. An overview of th discriptines within computer science such as networks, AI, robotics, graphics, and compute covered through relevant programming projects. The course sprinciples of compute organization, students will receive a foundation in programming focusing on C/C++. Fundamenta for computer science will be anterbaced robotics project that integrates the concept stage in the course. A lecture/lab course format will be employed to provide hands-on experie		Lecturer	Assosiate Prof Guliyev Maza	Assosiate Prof Guliyev Mazahim		
Phone number: 1494 55507 70 74 Lecture room/Schedule 111 Mehseiti Sireet, AZ1096 Baku, Azerbaijan (Neftchilar campus), room Course language English Type of the subject Saturday 13:00-14:00 Textbooks and additional materials <i>Textbooks</i> : I. C++ Programming: Program Design Including Data Structures, 6th Ed. D.S. Malik, Course Technology, 2011 . 2. Fundamentals of Computer programming with C++. Svetlin Nakov 2013 . 3. Introduction to Programming with C++. Svetlin Nakov 2013 . 3. Introduction to Programming with C++. Svetlin Nakov 2013 . 4. C: The Complete Reference, Herbert Schildt, McGrawHill, 2016 X Group discussions at seminars X Group discussions at seminars X Quizzes During the semester 20 Attendance 5 Midterm exam 30 Final exam 40 Final exam 40 Group is evidents will receive a foundation in programming from first principles of compute organization, students will receive a such as anetWorks, AI, robotics, graphics, and computer science such as networks, AI, robotics project that integrates the concept saught in the course. A lecture/lab course format will be employed to provide hands-on		E-mail:	mazahim.guliyev@gmail.com	mazahim.guliyev@gmail.com		
Lecture room/Schedule 11 Mehseii Street, AZ1096 Baku, Azerbaijan (Neftchilar campus), room Course language English Type of the Major subject Major Textbooks and additional materials Textbooks and additional 1. C++ Programming: Program Design Including Data Structures, 6th Ed. D.S. Malik, Course Technology, 2011 2. Fundamentals of Computer programming with C++. Svetlin Nakov 2013 3. Introduction to Programming with C++. Stellin Nakov 2013 3. Introduction to Programming with C++. Stellin Nakov 2013 4. C: The Complete Reference, Herbert Schildt, McGrawHill, 2016 Teaching methods X Group discussions at seminars X Quizzes During the semester 20 Quizzes During the semester 20 Quizzes During the semester 20 Attendance 5 Midterm exam 30 Final exam 40 Final 100 This course introduces students will receive a foundation in programming focusing on CC++. Fundamenta programming assignment and/or a team-based robotics, projet that integrates the concepts taught in the course. A lecture/lab course format will be employed to provide hands-on experience and active learning techniques		Phone number:	+994 55567 70 74			
Consultations Saturday 13:00 Course language English Type of the subject Major Textbooks and additional Textbooks: additional 1. C++ Programming: Program Design Including Data Structures, 6th Ed. D.S. Malik, Course Technology, 2011 2. Fundamentals of Computer programming with C++, Svetlin Nakov 2013 3. 3. Introduction to Programming with C++, Svetlin Nakov 2013 4. C: The Complete Reference, Herbert Schildt, McGrawHill, 2016 X Teaching Lecture X methods Group discussions at seminars X Assessment Components Date/ Deadline Percent (%) Active participation At each lesson 5 Quizzes Quizzes During the semester 20 Attendance 5 Midter me exam 40 100 Final exam 40 Einal exam Einal exam 40 Final exam integramming projects. The course will culminate in a comprehensive programming projects. The course will culminate in a comprehensive programming projects. The course will culminate in a comprehensive programming projects. The course will culminate in a com		Lecture room/Schedule	11 Mehseti Street, AZ1096 E	Baku, Azerbaijan (Neftchilar		
Constitutions Saturday 13:00-14:00 Course language English Type of the subject Major Textbooks and additional materials Textbooks: 1. C++ Programming: Program Design Including Data Structures, 6th Ed. D.S. Malik, Course Technology, 2011 2. Fundamentals of Computer programming with C++, 3rd Edition by Y. Daniel Liang, Pearson,2014 4. C: The Complete Reference, Herbert Schildt, McGrawHill, 2016 Teaching Lecture methods Group discussions at seminars Assessment Components Active participation At each lesson Attendance 5 Midterm exam 30 Final exam 40 Final exam 40 Final exam 40 Final exam 40 course architecture will be integrated throughout the course. Starting from first principles of compute architecture will be integrated throughout the course. Starting from first principles of compute architecture will be integrated robotics project that integrates the concepts taught in the course. A lecture/lab course format will be employed to provide hands-on experience and active learning coucepts along with current issues such as parallelism and embedded systems will be covered through relevant progr			campus), room			
Course tanguage English Type of the Major subject Textbooks and Textbooks and Textbooks: additional 1. C++ Programming: Program Design Including Data Structures, 6th Ed. D.S. Malik, Course Technology, 2011 2. Fundamentals of Computer programming with C++, Svetlin Nakov 2013 3. Introduction to Programming with C++, 3rd Edition by Y. Daniel Liang, Pearson, 2014 4. C: The Complete Reference, Herbert Schildt, McGrawHill, 2016 Teaching Lecture X methods Group discussions at seminars X Assessment Components Date/Deadline Quizzes During the semester 20 Attendance 30 Final Final exam 40 1000 This course introduces students to the field of computer science and engineering. An overview of th disciplines within computer science such as networks, AI, robotics, graphics, and compute organization, students will receive a foundation in programming from first principles of compute organizition, students will receive a foundation in programming clausing on C/C++. Fundamenta programming assignment and/or a team-based robotics project that integrates the concept study in the course. A lecture/lab course of compute organizition, students will be covered through relevant programming projects. The course will culminate in a comprehens	Course los ou o o	Consultations	Saturday 13:00 – 14:00			
Type of the Major subject Fextbooks and additional I. C++ Programming: Program Design Including Data Structures, 6th Ed. D.S. Malik, Course materials Textbooks: additional I. C++ Programming: Program Design Including Data Structures, 6th Ed. D.S. Malik, Course Technology, 2011 Production to Programming with C++. Svetlin Nakov 2013 3. Introduction to Programming with C++. Svetlin Nakov 2013 Introduction to Programming with C++. Svetlin Nakov 2013 Teaching Lecture X methods Group discussions at seminars X Assessment Components Date/Deadline Percent (%) Active participation At each lesson 5 Midterm exam 30 1000 Final 100 100 This course introduces students to the field of computer science and engineering. An overview of th disciplines within computer science such as networks, AI, robotics, graphics, and compute organization, students will receive a foundation in programming focusing on C/C++. Fundamenta programming concepts along with current issues such as parallelism and embedded systems will be covered through relevant programming projects. The course will culminate in a comprehensity programming assignment and/or a team-based robotics project that integrates the concepts taught in the course. A lecture/lab course its to introduce the programming langua	Type of the	Major				
Subject Textbooks and additional materials Textbooks: 1. C++ Programming: Program Design Including Data Structures, 6th Ed. D.S. Malik, Course Technology, 2011 2. Fundamentals of Computer programming with C++. Svetlin Nakov 2013 3. Introduction to Programming with C++. Svetlin Nakov 2013 3. Introduction to Programming with C++. Svetlin Nakov 2013 4. C: The Complete Reference, Herbert Schildt, McGrawHill, 2016 Teaching methods Components Date/ Deadline Assessment Components Date/ Deadline Quizzes During the semester 20 Attendance 5 Midterm exam 40 Final This course introduces students to the field of computer science and engineering. An overview of th disciplines within computer science such as networks, AI, robotics, graphics, and compute architecture will be integrated throughout the course. Starting from first principles of compute organization, students will receive a foundation in programming cousing on C/C++. Fundamenta programming concepts along with current issues such as parallelism and embedded systems will be covered through relevant programming projects. The course will culminate in a comprehensive programming assignment and/or a team-based robotics project that integrates the concepts taught in the course. A lecture/lab course format will be employed to provide hands-on experience and activ learning outcomes	subject	Major				
Additional 1. C++ Programming: Program Design Including Data Structures, 6th Ed. D.S. Malik, Course Technology, 2011 2. Fundamentals of Computer programming with C++, Svetlin Nakov 2013 3. Introduction to Programming with C++, 3rd Edition by Y. Daniel Liang,,Pearson,2014 4. C: The Complete Reference, Herbert Schildt, McGrawHill, 2016 Teaching methods Components Date/ Deadline Percent (%) Active participation At each lesson Quizzes During the semester 20 Attendance Midterm exam This course introduces students to the field of computer science and engineering. An overview of the disciplines within computer science such as networks, AI, robotics, graphics, and compute architecture will be integrated throughout the course. Starting from first principles of compute programming concepts along with current issues such as parallelism and embedded systems will be covered through relevant programming projects. The course will culminate in a comprehensive programming signment and/or a team-based robotics project that integrates the concepts taught in the course. A lecture/lab course format will be employed to provide hands-on experience and active learning techniques Course othe main objectives of this	Teythooks and	Tertbooks.				
materials 1	additional	1 C++ Programming: Program Design Including Data Structures 6th Ed. D.S. Malik Course				
2. Fundamentals of Computer programming with C++. Svetlin Nakov 2013 3. Introduction to Programming with C++. Svetlin Nakov 2013 3. Introduction to Programming with C++. Svetlin Nakov 2013 3. Introduction to Programming with C++. Svetlin Nakov 2013 3. Introduction to Programming with C++. Svetlin Nakov 2013 3. Introduction to Programming with C++. Svetlin Nakov 2013 3. Introduction to Programming with C++. Svetlin Nakov 2013 Teaching methods X Group discussions at seminars X Assessment Components Date/Deadline Active participation At each lesson 5 Quizzes During the semester 20 Attendance 5 5 Midterm exam 30 100 This course introduces students to the field of computer science and engineering. An overview of the disciplines within computer science such as networks, AI, robotics, graphics, and compute organization, students will receive a foundation in programming focusing on C/C++. Fundamenta programming concepts along with current issues such as parallelism and embedded systems will be covered through relevant programming projects. The course will culminate in a comprehensive programming assignment and/or a team-based robotics project that integrates the concepts taught in the course. A lecture/lab course format will be employed to provide hands-on experience and active learning outcomes Upon completion	materials	Technology, 2011				
3. Introduction to Programming with C++, 3rd Edition by Y. Daniel Liang, Pearson, 2014 4. C: The Complete Reference, Herbert Schildt, McGrawHill, 2016 Teaching from the semantic second secon	materials	2. Fundamentals of Computer prog	ramming with C++. Svetlin Nakov	2013		
4. C: The Complete Reference, Herbert Schildt, McGrawHill, 2016 X Teaching methods Lecture X Assessment Components Date/ Deadline Percent (%) Active participation At each lesson 5 Quizzes During the semester 20 Attendance 5 5 Midterm exam 30 5 Final exam 40 100 Course This course introduces students to the field of computer science and engineering. An overview of the disciplines within computer science such as networks, AI, robotics, graphics, and compute architecture will be integrated throughout the course. Starting from first principles of compute organization, students will receive a foundation in programming foreignes of compute organization, students will receive a foundation in programming foreignes will be covered through relevant programming projects. The course will culminate in a comprehensive programming techniques Course The main objectives of this course is to introduce the programming language and aware the students about programming paradigm. and to focus concept and methodology of programming objectives about programming paradigm. and to focus concept and methodology of programming Upon completion of this course is to introduce the programming language and aware the students objectives about programming paradigm. and to focus concept and methodology of programming Upon completion of t		 Introduction to Programming with C++ 3rd Edition by Y Daniel Liang Pearson 2014 				
Teaching methods Lecture X Assessment Group discussions at seminars X Assessment Components Date/Deadline Percent (%) Active participation At each lesson 5 Quizzes During the semester 20 Attendance 5 5 Midtern exam 30 5 Final exam 40 100 Final exam 40 100 This course introduces students to the field of computer science and engineering. An overview of the disciplines within computer science such as networks, AI, robotics, graphics, and compute organization, students will receive a foundation in programming forusing on C/C++. Fundamenta programming concepts along with current issues such as parallelism and embedded systems will be covered through relevant programming projects. The course will culminate in a comprehensive programming assignment and/or a team-based robotics project that integrates the concepts taught in the course. A lecture/lab course format will be employed to provide hands-on experience and active learning between the student's personal learning goals. Course objectives of this course students will be able to: 1.Define and discuss the disciplines of computer science with the intention of choosing future courses that are applicable to the student's personal learning goals. 2. Understand the key hardware components in		4. C: The Complete Reference, He	rbert Schildt, McGrawHill, 2016			
methods Group discussions at seminars X Assessment Components Date/ Deadline Percent (%) Active participation At each lesson 5 Quizzes During the semester 20 Attendance 5 Midterm exam 30 Final exam 440 Final 40 Final exam 40 Final 7 Course This course introduces students to the field of computer science and engineering. An overview of the disciplines within computer science such as networks, AI, robotics, graphics, and compute organization, students will receive a foundation in programming from first principles of compute organization, students will receive a foundation in programming focusing on C/C++. Fundamenta programming concepts along with current issues such as parallelism and embedded systems will be covered through relevant programming projects. The course will culminate in a comprehensive programming techniques Course The main objectives of this course is to introduce the programming language and aware the students about programming paradigm. and to focus concept and methodology of programming objectives about programming paradigm. and to focus concept and methodology of programming outcomes Upon completion of this course students will be able to: 1.Define and discuss the disciplines of computer science with the intention o	Teaching	Lecture		Х		
Assessment Components Date/ Deadline Percent (%) Active participation Acteach lesson 5 Quizzes During the semester 20 Attendance 5 Midterm exam 30 Final exam 40 Final 100 This course introduces students to the field of computer science and engineering. An overview of the disciplines within computer science such as networks, AI, robotics, graphics, and compute architecture will be integrated throughout the course. Starting from first principles of compute architecture will be integrated throughout the course. Starting from first principles of compute organization, students will receive a foundation in programming focusing on C/C++. Fundamenta programming concepts along with current issues such as parallelism and embedded systems will be covered through relevant programming projects. The course will culminate in a comprehensive programming assignment and/or a team-based robotics project that integrates the concepts taught in the course. A lecture/lab course is to introduce the programming language and aware the students objectives about programming paradigm. and to focus concept and methodology of programming Learning outcomes Upon completion of this course students will be able to: 1.Define and discuss the disciplines of computer science with the intention of choosing future courses that are applicable to the student's personal learning goals. 2. Understand the key hardware components in a modern computer system and how software is mapp	methods	Group discussions at seminars		Х		
Active participation At each lesson 5 Quizzes During the semester 20 Attendance 5 Midterm exam 30 Final exam 40 Final 100 This course introduces students to the field of computer science and engineering. An overview of the disciplines within computer science such as networks, AI, robotics, graphics, and compute organization, students will receive a foundation in programming focusing on C/C++. Fundamenta programming concepts along with current issues such as parallelism and embedded systems will be covered through relevant programming projects. The course will culminate in a comprehensive programming assignment and/or a team-based robotics project that integrates the concepts taught in the course. A lecture/lab course format will be employed to provide hands-on experience and active learning techniques Course The main objectives of this course is to introduce the programming language and aware the students about programming paradigm. and to focus concept and methodology of programming Learning Upon completion of this course students will be able to: 1. Define and discuss the disciplines of computer science with the intention of choosing future courses that are applicable to the student's personal learning goals. 2. Understand the key hardware components in a modern computer system and how software is mapped to the HW. 3. Use a computer to solve problems by developing simple algorithms and then implement them using a specific programmi	Assessment	Components	Date/ Deadline	Percent (%)		
Quizzes During the semester 20 Attendance 5 Midterm exam 30 Final exam 40 Final course Final description This course introduces students to the field of computer science and engineering. An overview of the disciplines within computer science such as networks, AI, robotics, graphics, and compute architecture will be integrated throughout the course. Starting from first principles of computer organization, students will receive a foundation in programming focusing on C/C++. Fundamenta programming concepts along with current issues such as parallelism and embedded systems will be covered through relevant programming projects. The course will culminate in a comprehensive programming assignment and/or a team-based robotics project that integrates the concepts taught in the course. A lecture/lab course format will be employed to provide hands-on experience and active learning techniques Course The main objectives of this course is to introduce the programming language and aware the students about programming paradigm. and to focus concept and methodology of programming Learning Upon completion of this course students will be able to: 1.Define and discuss the disciplines of computer science with the intention of choosing future courses that are applicable to the student's personal learning goals. 2. Understand the key hardware components in a modern computer system and how software is mapped to the HW. 3. Use a computer to solve problems by developing simple algorithms and then implement them		Active participation	At each lesson	5		
Attendance 5 Midterm exam 30 Final exam 40 Final 100 Course description This course introduces students to the field of computer science and engineering. An overview of the disciplines within computer science such as networks, AI, robotics, graphics, and compute organization, students will receive a foundation in programming focusing on C/C++. Fundamenta programming concepts along with current issues such as parallelism and embedded systems will be covered through relevant programming projects. The course will culminate in a comprehensive programming assignment and/or a team-based robotics project that integrates the concepts taught in the course. A lecture/lab course format will be employed to provide hands-on experience and active learning techniques Course The main objectives of this course is to introduce the programming language and aware the students about programming paradigm. and to focus concept and methodology of programming Learning Upon completion of this course students will be able to: 1.Define and discuss the disciplines of computer science with the intention of choosing future courses that are applicable to the student's personal learning goals. 2. Understand the key hardware components in a modern computer system and how software is mapped to the HW. 3. Use a computer to solve problems by developing simple algorithms and then implement them using a specific programming language 4. Implement key algorithms within the field 5. Understand and determine the computational		Quizzes	During the semester	20		
Midterm exam 30 Final exam 40 Final exam 40 Final 100 Course disciplines within computer science such as networks, AI, robotics, graphics, and compute architecture will be integrated throughout the course. Starting from first principles of compute organization, students will receive a foundation in programming focusing on C/C++. Fundamenta programming concepts along with current issues such as parallelism and embedded systems will be covered through relevant programming projects. The course will culminate in a comprehensive programming assignment and/or a team-based robotics project that integrates the concepts taught in the course. A lecture/lab course format will be employed to provide hands-on experience and active learning techniques Course The main objectives of this course is to introduce the programming language and aware the students about programming paradigm. and to focus concept and methodology of programming Learning Upon completion of this course students will be able to: 1. Define and discuss the disciplines of computer science with the intention of choosing future courses that are applicable to the student's personal learning goals. 2. Understand the key hardware components in a modern computer system and how software is mapped to the HW. 3. Use a computer to solve problems by developing simple algorithms and then implement them using a specific programming language 4. Implement key algorithms within the field 5. Understand and determine the computational complexity of		Attendance		5		
Final exam 40 Final 100 Course description This course introduces students to the field of computer science and engineering. An overview of the disciplines within computer science such as networks, AI, robotics, graphics, and compute organization, students will receive a foundation in programming focusing on C/C++. Fundamenta programming concepts along with current issues such as parallelism and embedded systems will be covered through relevant programming projects. The course will culminate in a comprehensive programming assignment and/or a team-based robotics project that integrates the concepts taught in the course. A lecture/lab course format will be employed to provide hands-on experience and active learning techniques Course The main objectives of this course is to introduce the programming language and aware the students about programming paradigm. and to focus concept and methodology of programming Learning Upon completion of this course students will be able to: 1.Define and discuss the disciplines of computer science with the intention of choosing future courses that are applicable to the student's personal learning goals. 2. Understand the key hardware components in a modern computer system and how software is mapped to the HW. 3. Use a computer to solve problems by developing simple algorithms and then implement them using a specific programming language 4. Implement key algorithms within the field 5. Understand and determine the computitional complexity of simple algorithms		Midterm exam		30		
Final 100 Course description This course introduces students to the field of computer science and engineering. An overview of the disciplines within computer science such as networks, AI, robotics, graphics, and compute organization, students will receive a foundation in programming focusing on C/C++. Fundamenta programming concepts along with current issues such as parallelism and embedded systems will be covered through relevant programming projects. The course will culminate in a comprehensive programming assignment and/or a team-based robotics project that integrates the concepts taught in the course. A lecture/lab course format will be employed to provide hands-on experience and active learning techniques Course The main objectives of this course is to introduce the programming language and aware the students about programming paradigm. and to focus concept and methodology of programming Learning Upon completion of this course students will be able to: 1.Define and discuss the disciplines of computer science with the intention of choosing future courses that are applicable to the student's personal learning goals. 2. Understand the key hardware components in a modern computer system and how software is mapped to the HW. 3. Use a computer to solve problems by developing simple algorithms and then implement them using a specific programming language 4. Implement key algorithms within the field 5. Understand and determine the computational complexity of simple algorithms		Final exam		40		
Course description This course introduces students to the field of computer science and engineering. An overview of the disciplines within computer science such as networks, AI, robotics, graphics, and compute architecture will be integrated throughout the course. Starting from first principles of compute organization, students will receive a foundation in programming focusing on C/C++. Fundamenta programming concepts along with current issues such as parallelism and embedded systems will be covered through relevant programming projects. The course will culminate in a comprehensive programming assignment and/or a team-based robotics project that integrates the concepts taught in the course. A lecture/lab course format will be employed to provide hands-on experience and active learning techniques Course objectives The main objectives of this course is to introduce the programming language and aware the students about programming paradigm. and to focus concept and methodology of programming Learning outcomes Upon completion of this course students will be able to: 1.Define and discuss the disciplines of computer science with the intention of choosing future courses that are applicable to the student's personal learning goals. 2. Understand the key hardware components in a modern computer system and how software is mapped to the HW. 3. Use a computer to solve problems by developing simple algorithms and then implement them using a specific programming language 4. Implement key algorithms within the field 5. Understand and determine the computational complexity of simple algorithms		Final		100		
Course description disciplines within computer science such as networks, AI, robotics, graphics, and compute architecture will be integrated throughout the course. Starting from first principles of compute organization, students will receive a foundation in programming focusing on C/C++. Fundamenta programming concepts along with current issues such as parallelism and embedded systems will be covered through relevant programming projects. The course will culminate in a comprehensive programming assignment and/or a team-based robotics project that integrates the concepts taught in the course. A lecture/lab course format will be employed to provide hands-on experience and active learning techniques Course objectives The main objectives of this course is to introduce the programming language and aware the students about programming paradigm. and to focus concept and methodology of programming Learning outcomes Upon completion of this course students will be able to: 1.Define and discuss the disciplines of computer science with the intention of choosing future courses that are applicable to the student's personal learning goals. 2. Understand the key hardware components in a modern computer system and how software is mapped to the HW. 3. Use a computer to solve problems by developing simple algorithms and then implement them using a specific programming language 4. Implement key algorithms within the field 5. Understand and determine the computational complexity of simple algorithms		This course introduces students to the	he field of computer science and e	ngineering. An overview of the		
description architecture will be integrated throughout the course. Starting from first principles of compute organization, students will receive a foundation in programming focusing on C/C++. Fundamenta programming concepts along with current issues such as parallelism and embedded systems will be covered through relevant programming projects. The course will culminate in a comprehensive programming assignment and/or a team-based robotics project that integrates the concepts taught in the course. A lecture/lab course format will be employed to provide hands-on experience and active learning techniques Course The main objectives of this course is to introduce the programming language and aware the students about programming paradigm. and to focus concept and methodology of programming Learning Upon completion of this course students will be able to: 1.Define and discuss the disciplines of computer science with the intention of choosing future courses that are applicable to the student's personal learning goals. 2. Understand the key hardware components in a modern computer system and how software is mapped to the HW. 3. Use a computer to solve problems by developing simple algorithms and then implement them using a specific programming language 4. Implement key algorithms within the field 5. Understand and determine the computational complexity of simple algorithms	Course	disciplines within computer science such as networks, AI, robotics, graphics, and computer				
organization, students will receive a foundation in programming focusing on C/C++. Fundamenta programming concepts along with current issues such as parallelism and embedded systems will be covered through relevant programming projects. The course will culminate in a comprehensive programming assignment and/or a team-based robotics project that integrates the concepts taught in the course. A lecture/lab course format will be employed to provide hands-on experience and active learning techniques Course The main objectives of this course is to introduce the programming language and aware the students about programming paradigm. and to focus concept and methodology of programming Learning Upon completion of this course students will be able to: 1.Define and discuss the disciplines of computer science with the intention of choosing future courses that are applicable to the student's personal learning goals. 2. Understand the key hardware components in a modern computer system and how software is mapped to the HW. 3. Use a computer to solve problems by developing simple algorithms and then implement them using a specific programming language 4. Implement key algorithms within the field 5. Understand and determine the complexity of simple algorithms	description	architecture will be integrated thro	bughout the course. Starting from	n first principles of computer		
programming concepts along with current issues such as parallelism and embedded systems will be covered through relevant programming projects. The course will culminate in a comprehensive programming assignment and/or a team-based robotics project that integrates the concepts taught in the course. A lecture/lab course format will be employed to provide hands-on experience and active learning techniques Course objectives The main objectives of this course is to introduce the programming language and aware the students about programming paradigm. and to focus concept and methodology of programming Learning outcomes Upon completion of this course students will be able to: 1.Define and discuss the disciplines of computer science with the intention of choosing future courses that are applicable to the student's personal learning goals. 2. Understand the key hardware components in a modern computer system and how software is mapped to the HW. 3. Use a computer to solve problems by developing simple algorithms and then implement them using a specific programming language 4. Implement key algorithms within the field 5. Understand and determine the computational complexity of simple algorithms		organization, students will receive a foundation in programming focusing on C/C++. Fundamental				
covered through relevant programming projects. The course will culminate in a comprehensive programming assignment and/or a team-based robotics project that integrates the concepts taught in the course. A lecture/lab course format will be employed to provide hands-on experience and active learning techniques Course The main objectives of this course is to introduce the programming language and aware the students about programming paradigm. and to focus concept and methodology of programming Learning Upon completion of this course students will be able to: 1.Define and discuss the disciplines of computer science with the intention of choosing future courses that are applicable to the student's personal learning goals. 2. Understand the key hardware components in a modern computer system and how software is mapped to the HW. 3. Use a computer to solve problems by developing simple algorithms and then implement them using a specific programming language 4. Implement key algorithms within the field 5. Understand and determine the computational complexity of simple algorithms 6. Write computer programming and iteration and the intertion of choosen function of the student is a modern complexity of simple algorithms		programming concepts along with current issues such as parallelism and embedded systems will be				
Image: Programming assignment and/or a team-based robotics project that integrates the concepts tadgit in the course. A lecture/lab course format will be employed to provide hands-on experience and active learning techniques Course The main objectives of this course is to introduce the programming language and aware the students about programming paradigm. and to focus concept and methodology of programming Learning Upon completion of this course students will be able to: 1.Define and discuss the disciplines of computer science with the intention of choosing future courses that are applicable to the student's personal learning goals. 2. Understand the key hardware components in a modern computer system and how software is mapped to the HW. 3. Use a computer to solve problems by developing simple algorithms and then implement them using a specific programming language 4. Implement key algorithms within the field 5. Understand and determine the computational complexity of simple algorithms 6. Write computer means with a field		covered through relevant programming projects. The course will culminate in a comprehensive				
Course The main objectives of this course is to introduce the programming language and aware the students about programming paradigm. and to focus concept and methodology of programming Learning Upon completion of this course students will be able to: 1.Define and discuss the disciplines of computer science with the intention of choosing future courses that are applicable to the student's personal learning goals. 2. Understand the key hardware components in a modern computer system and how software is mapped to the HW. 3. Use a computer to solve problems by developing simple algorithms and then implement them using a specific programming language 4. Implement key algorithms within the field 5. Understand and determine the computational complexity of simple algorithms 6. Write commuter mercarement with a contractional complexity of simple algorithms		programming assignment and/or a team-based robotics project that integrates the concepts taught in				
Course The main objectives of this course is to introduce the programming language and aware the students about programming paradigm. and to focus concept and methodology of programming Learning outcomes Upon completion of this course students will be able to: 1.Define and discuss the disciplines of computer science with the intention of choosing future courses that are applicable to the student's personal learning goals. 2. Understand the key hardware components in a modern computer system and how software is mapped to the HW. 3. Use a computer to solve problems by developing simple algorithms and then implement them using a specific programming language 4. Implement key algorithms within the field 5. Understand and determine the computational complexity of simple algorithms 6. Write computer to programming the programming language		the course. A lecture/lab course format will be employed to provide hands-on experience and active				
course The main objectives of this course is to introduce the programming inigitige and aware the statents about programming paradigm. and to focus concept and methodology of programming Learning outcomes Upon completion of this course students will be able to: 1.Define and discuss the disciplines of computer science with the intention of choosing future courses that are applicable to the student's personal learning goals. 2. Understand the key hardware components in a modern computer system and how software is mapped to the HW. 3. Use a computer to solve problems by developing simple algorithms and then implement them using a specific programming language 4. Implement key algorithms within the field 5. Understand and determine the computational complexity of simple algorithms 6. Write computer programming using aorditional complexity of simple algorithms	Course	The main objectives of this course is to introduce the programming language and aware the students				
Learning upon completion of this course students will be able to: 1.Define and discuss the disciplines of computer science with the intention of choosing future courses that are applicable to the student's personal learning goals. 2. Understand the key hardware components in a modern computer system and how software is mapped to the HW. 3. Use a computer to solve problems by developing simple algorithms and then implement them using a specific programming language 4. Implement key algorithms within the field 5. Understand and determine the computational complexity of simple algorithms	objectives	about programming paradigm, and to focus concept and methodology of programming				
Outcomes Upon completion of this course students will be able to: 1.Define and discuss the disciplines of computer science with the intention of choosing future courses that are applicable to the student's personal learning goals. 2. Understand the key hardware components in a modern computer system and how software is mapped to the HW. 3. Use a computer to solve problems by developing simple algorithms and then implement them using a specific programming language 4. Implement key algorithms within the field 5. Understand and determine the computational complexity of simple algorithms	Learning			P. 8		
 Upon completion of this course students will be able to: 1.Define and discuss the disciplines of computer science with the intention of choosing future courses that are applicable to the student's personal learning goals. 2. Understand the key hardware components in a modern computer system and how software is mapped to the HW. 3. Use a computer to solve problems by developing simple algorithms and then implement them using a specific programming language 4. Implement key algorithms within the field 5. Understand and determine the computational complexity of simple algorithms 						
 Define and discuss the disciplines of computer science with the intention of choosing future courses that are applicable to the student's personal learning goals. Understand the key hardware components in a modern computer system and how software is mapped to the HW. Use a computer to solve problems by developing simple algorithms and then implement them using a specific programming language Implement key algorithms within the field Understand and determine the computational complexity of simple algorithms 	outcomes	Upon completion of this course students will be able to:				
 1.Define and discuss the disciplines of computer science with the intention of choosing future courses that are applicable to the student's personal learning goals. 2. Understand the key hardware components in a modern computer system and how software is mapped to the HW. 3. Use a computer to solve problems by developing simple algorithms and then implement them using a specific programming language 4. Implement key algorithms within the field 5. Understand and determine the computational complexity of simple algorithms 						
 that are applicable to the student's personal learning goals. 2. Understand the key hardware components in a modern computer system and how software is mapped to the HW. 3. Use a computer to solve problems by developing simple algorithms and then implement them using a specific programming language 4. Implement key algorithms within the field 5. Understand and determine the computational complexity of simple algorithms 		1 Define and discuss the disciplines of computer science with the intention of choosing future courses				
 2. Understand the key hardware components in a modern computer system and how software is mapped to the HW. 3. Use a computer to solve problems by developing simple algorithms and then implement them using a specific programming language 4. Implement key algorithms within the field 5. Understand and determine the computational complexity of simple algorithms 6. Write computer programs using conditional and iterative structures functional decomposition and 		that are applicable to the student's personal learning goals.				
 mapped to the HW. 3. Use a computer to solve problems by developing simple algorithms and then implement them using a specific programming language 4. Implement key algorithms within the field 5. Understand and determine the computational complexity of simple algorithms 6. Write computer programs using conditional and iterative structures functional decomposition and 		2. Understand the key hardware components in a modern computer system and how software is				
 3. Use a computer to solve problems by developing simple algorithms and then implement them using a specific programming language 4. Implement key algorithms within the field 5. Understand and determine the computational complexity of simple algorithms 6. Write computer programs using conditional and iterative structures functional decomposition, and 		mapped to the HW.				
 a specific programming language 4. Implement key algorithms within the field 5. Understand and determine the computational complexity of simple algorithms 6. Write computer programs using conditional and iterative structures functional decomposition, and 		3. Use a computer to solve problems by developing simple algorithms and then implement them using				
 4. Implement key algorithms within the field 5. Understand and determine the computational complexity of simple algorithms 6. Write computer programs using conditional and iterative structures functional decomposition, and 		a specific programming language				
5. Understand and determine the computational complexity of simple algorithms		4. Implement key algorithms within the field				
6 Write computer programs using conditional and iterative structures functional decomposition and		5. Understand and determine the computational complexity of simple algorithms				
0. write computer programs using conditional and iterative structures, functional decomposition, and		6. Write computer programs using conditional and iterative structures, functional decomposition, and				
basic parallelization techniques		basic parallelization techniques				
7. Select an appropriate basic data structure (e.g. arrays) and access methods (e.g. pointers)		7. Select an appropriate basic data structure (e.g. arrays) and access methods (e.g. pointers)				
8. Understand basic object-oriented principles.		8. Understand basic object-oriented principles.				
9. Design non-trivial embedded software to control a robot to navigate and interact within a controlled		9. Design non-trivial embedded software to control a robot to navigate and interact within a controlled				
environment Pulse Lesson organization	Dulos	Lesson organization				
(Educational General information on the subject will be provided for the students during lectures	(Educational	General information on the subject wi	ll be provided for the students dur	ing lectures.		

policy a behavio	and or)	Student's knowledge on the previous topics will be evaluated and new topic will be of visual aids during seminars. Student's knowledge level will be tested oraly and before midterm and final exams. Submission of the individual works by the end of co Attendance Participation of students at all classis is important. Students should inform dean's of lessons for particular reasons (illness, family issues and etc.). Students, missing relessons, are not allowed to take the exam. Lates Those students who are late for lessons for more than 15 minutes are not allowed to lesson. Despite this, the student is allowed to take part in the second part of the lesson Quizzes Those students who have informed the teacher and the dean's office about missing the for particular reasons, are allowed to take the quiz next week. Exams All the issues related to the participation and admission to the exam are regulated by to Topics of midterm and final exams are provided for the students before the exams. Violation of the rules of the exams Disrupting the quiz and taking copy during midterm and final exams is forbidden. O student who do not follow these rules are canceled and the students are expelled getting 0 (zero). The rule for completing the course In accordance with the University rules the overall success rate to complete the cours or above. The students who failed the exam would be to take this subject next semester Rules of conduct for Students Disruption of the lesson and not following ethical norms during the lesson, as well as discussions by the students without permission and using mobile phones is forbidden.	explained by mins l in written forms urse is obligatory. fice about missing nore than 25% of o participate at the he quiz in advance the faculty dean. The questions of Quiz papers of the from the quiz by rse should be 60% er or next year. conduction of the
Week	Dates (planned)	Subject topics	Textbook/ Assignments
1	(F)	Introduction Generation and Classification of Computers Basic Organization of a Computer–Number System –Binary – Decimal – Conversion – Problems.	[2] p. 2-16
2		Algorithm Need for logical analysis and thinking Pseudo code Flow Chart. Problem Solving	[1] p. 4-20 [1] p.30-52 [1] p. 24-28 [1] p. 67-81
3		<i>C Programming Basics</i> Problem formulation– Introduction to 'C' programming –fundamentals – structure of a 'C' program – compilation and linking processes	[2] p. 31-57 [2] p. 76-85
4		C Programming Basics Constants, Variables – Data Types Expressions using operators in 'C'	[2] p. 113-146 [2] p. 153-155
5		<i>C Programming Basics</i> Managing Input and Output operations – Decision Making and Branching– solving simple scientific and statistical problems.	[1] p. 216-240 [3] p. 124-148
		Quiz 1(Lec1-Lec4)	[1] p. 242-251
6		Public holiday	
7		C Programming Basics Expressions using operators in 'C' – Managing Input and Output operations – Decision Making and Branching– solving simple scientific and statistical problems.	[2] p. 173-198 [2] p. 201-208 [2] p. 201-208
8		Looping statements Quiz 2(Lec5-Lec6)	[2] p. 271-300
9		Mid term exam	

10	Arrays and Strings Arrays – Initialization – Declaration – One dimensional and Two dimensional arrays	[2] p. 384-422 [2] p. 425-427
11	Arrays and Strings String- String operations – String Arrays - Simple programs Sorting- searching – matrix operations	[2] p. 565-588 [2] p. 596-600
12	<i>Functions and Pointers</i> Function – definition of function – Declaration of function – Pass by value – Pass by reference – Recursion <i>Quiz 3(Lec9-Lec10)</i>	[2] p. 602-635
13	<i>Functions and Pointers</i> Pointers – Definition – Initialization – Pointers arithmetic – Pointers and arrays- Example Problems.	[2] p. 764-788 [2] p. 801-805
14	Structures and Unions Introduction – need for structure data type – structure definition – Structure declaration - Structure within a structure – Union – Programs using structures and Unions – Storage classes, Pre-processor directives	[2] p. 807-825 [2] p. 845-850
15	Recap of all covered material Quiz 4(Lec11-Lec13)	
	Final Exam	