Course catalogue

Computer Science

academic year 2020/21

BJELOVAR UNIVERSITY OF APPLIED SCIENCES

GENERAL INFORMATION							
Course title	**Introduction to Informatics**						
Programme of study	Undergraduate professional programme of study in Computer Science						
Course status	Compulsory						
Year	1st						
Semester	1st						
Calculation of ECTS credits	Lectures	Practical sessions		Preparation for practical sessions	Seminar paper	Self-study	Total
		APS	LPS				
	30	0	30	30	0	90	180
ECTS credits and forms of instruction	ECTS student workload coefficient				7		
	Contact hours (L+PS+S)				L	PS	S
					30	30	

1. COURSE DESCRIPTION

1.1. Course objectives

1. Getting familiarised with the development of information technologies.
2. Learning the architecture, operation principles of personal computers and the operating system *Windows* and applying them in practice.
3. Learning the fundamentals of computer networks and network connection of personal computers and applying them in practice.
4. Learning the software tool *Office* and applying it in office work and business environment.
5. Learning the fundamentals of PC maintenance and security.

1.2. Prerequisites

None.

1.3. Expected learning outcomes

Upon completion of the course students will be able to:

1. recognise and describe assemblies and tools of personal computers and computer network,
2. describe installation and adjustment procedures of: the operating system *Windows*, e-mail, administration tools for controlling PC operation and antivirus computer protection,
3. apply their knowledge of the Microsoft software tool *Office* (*Word* – text compiling and editing, *Excel* – table calculations, *PowerPoint* – information presentation and *Access* – relation database) in practical preparation of seminar papers, thesis and for technical and administrative needs in the business sector.

1.4. Course content

1. The notion of computer literacy

Innovations that caused the development of information technologies.

2. Assemblies and architecture of personal computers

Von Neumman's model of a digital computer. CPU processor. Memory (RAM, ROM-BIOS, CASHE). Input units (hard disk, CD, DVD, modem, USB, keyboard, mouse, touchpad, scanner, web-camera, digital camera). Output units (monitor, projector, printers, plotters).

3. Operating systems

Early beginnings and development. Operating systems *Windows* and *Linux* – characteristics and installation. Operating environment in operating systems – GUI, Kernel.

4. Computer networks and network connection between computers

Network services. LAN adjustment. Internet access, installation and adjustment. Operation and usage of basic network services: e-mail, finding information on the Internet, distance operation, FTP server.

5. Maintenance and security of the operating system *Windows*

Administration tools. Antivirus protection. Firewall.

6. MS Office tools

Word. Excel. Access. PowerPoint, Outlook.

1.5. Forms of instruction	☒ lectures ☐ seminars and workshops ☒ practical sessions ☐ distance learning ☐ field work	☐ individual work ☐ multimedia and web ☒ laboratory sessions ☐ mentorship ☐ other _____
1.6. Comments		

1.7. Students' obligations

2 preliminary exams
15 laboratory sessions

1.8. Students' performance monitoring

Class attendance	**0.5**	Class participation		Seminar paper		Experimental work	0
Written exam	**2.5**	Oral exam	**1**	Essay		Research	
Project		Continuous assessment	**1**	Class report		Practical work	**2**
Portfolio							

1.9. Grading and evaluation of student work in class and in final exam

Activity in laboratory sessions: 40 points
Two preliminary exams / exam: 60 points (a minimum of 20 points is to be achieved),
1st preliminary exam: 30 points (a minimum of 10 points is to be achieved),
2nd preliminary exam: 30 points (a minimum of 10 points is to be achieved).

A maximum of 100 points can be scored in the course. Students pass the written exam if they have achieved a total of 50 points doing their laboratory practice and preliminary exams. Students who have passed the written exam by passing the preliminary exams are exempt from taking the oral exam, except if they wish to improve their grade.

Grading and evaluation of students' work upon completion of the course:
Written exam: 100 points
Oral exam

Students pass the written exam if they have scored 50 points. After they have passed the written exam, students take the oral exam.

WRITTEN EXAM GRADING:
[50, 63> points – sufficient (2)
[63, 75> points – good (3)
[75, 87> points – very good (4)
[87, 100] points – excellent (5)

1.10. Required reading (at the time of proposing the study programme)

1. Robert Herčeki: Presentations for lectures and practical sessions in the course “Introduction to Informatics”, Technical College in Bjelovar.
2. Šimović, Maletić, Afrić: *Osnove informatike*, Zagreb, 2010

1.11. Further reading (at the time of proposing the study programme)

1.12. Number of copies of required reading compared to the number of students currently taking the course

Title	*Number of copies*	*Number of students*
Robert Herčeki: Presentations for lectures and practical sessions in the course “Introduction to Informatics”, Technical College in Bjelovar.	0, available *online*	30
Šimović, Maletić, Afrić: Osnove informatike, Zagreb 2010	6	30

1.13. Methods of quality monitoring aimed at ensuring the acquisition of output knowledge, skills and competences

Conducting student surveys and evaluating the data in accordance with the results of the Student Survey Committee and the analyses of exam results in accordance with the data from the ISVU data warehouse.

1.14. Possibility of course instruction in a foreign language

English

GENERAL INFORMATION							
Course title	**Introduction to Programming**						
Programme of study	Undergraduate professional programme of study in Computer Science						
Course status	Compulsory						
Year	1st						
Semester	1st						
Calculation of ECTS credits	Lectures	Practical sessions		Preparation for practical sessions	Seminar paper	Self-study	Total
		APS	LPS				
	30	0	30	30	0	120	210
ECTS credits and forms of instruction	ECTS student workload coefficient				7		
	Contact hours (L+PS+S)				L	PS	S
					30	30	

1. COURSE DESCRIPTION

1.1. Course objectives

1. Developing the ability of algorithmic approach to problem solving.
2. Acquiring basic knowledge of programming in *Python*.
3. Developing logical thought processes in the problem-solving process.

1.2. Prerequisites

None.

1.3. Expected learning outcomes

Upon completion of the course students will be able to:

1. describe the steps of solving a problem by programming.
2. use the programming environment and the implementation of a program on a computer in the programming language *Python.*
3. explain handling different types of data during programming.
4. explain the control during program execution.
5. write a simple program.

1.4. Course content

1. Programming languages and programming
The history of programming languages. Potential applications of programming. Program development. Thinking like a programmer.
2.Programming language *Python*
Introduction to *Python*. Setting up *Python* on *Windows*. Programming a "Hello World" program. *Python* syntax.
3. Handling simple data
Standard data types. Variables. Display of numbers. Arithmetic operators. Operators of comparison. Assignment operators. Logical operators. Bitwise operators. Priority of operators. Data type conversions. Mathematical functions. Simple input-output functions.
4. Control during program execution
Conditional execution. Conditional commands with one sentence and several sentences. Nesting of

conditional commands. Programming loop. While loop. For loop. Break and continue commands. Pass command.

5.Strings

String data. Special signs in strings. Formatting strings. Operations on strings.

6.Collections of objects

Consecutive collections. Lists. Operations on lists. Matrices. N-tuples. Operations on n-tuples. Associative collections. Dictionaries. Operations on dictionaries. Sets. Operations on sets.

7.Funkcije

Definition of a function. Calling a function. Function arguments. Return values of functions. Variable scope.

8.Files

Operations with files. Reading and writing textual and binary files.

1.5. Forms of instruction	☒ lectures ☐ seminars and workshops ☒ practical sessions ☐ distance learning ☐ field work	☐ individual work ☐ multimedia and web ☒ laboratory sessions ☐ mentorship ☐ other _____
1.6. Comments		

1.7. Students' obligations

2 preliminary exams
2 homework assignments
15 laboratory sessions

1.8. Students' performance monitoring

Class attendance	**0.5**	Class participation		Seminar paper		Experimental work	0
Written exam	**2.5**	Oral exam	**1**	Essay		Research	
Project		Continuous assessment	**1**	Class report		Practical work	**2**
Portfolio							

1.9. Grading and evaluation of student work in class and in final exam

Homework assignments: 5 x 2 points = 10 points
Activity in laboratory sessions: 30 points
Two preliminary exams: 60 points (a minimum of 20 points is required),
1st preliminary exam: 30 points (a minimum of 10 points is required),
2nd preliminary exam: 30 points (a minimum of 10 points is required).

A maximum of 100 points can be scored in the course. Students pass the written exam if they have achieved a total of 50 points doing their homework assignments, laboratory practice and preliminary exams. Students who have passed the written exam by passing the preliminary exams are exempt from taking the oral exam, except if they wish to improve their grade.

Grading and evaluation of students' work upon completion of the course:
Written exam: 100 points
Oral exam

Students pass the written exam if they have scored 50 points. After they have passed the written exam, students take the oral exam.

WRITTEN EXAM GRADING:
[50, 63> points – sufficient (2)
[63, 75> points – good (3)

[75, 87> points – very good (4)
[87, 100] points – excellent (5)

1.10. Required reading (at the time of proposing the study programme)

- Alan Mutka: Presentations for lectures and practical sessions in the course "Introduction to Programming", Technical College in Bjelovar
- Zoran Kalafatić, Antonio Pošćić, Siniša Šegvić, Julijan Šribar: *Python za znatiželjne*, Element, Zagreb, 2016

1.11. Further reading (at the time of proposing the study programme)

- Michael Dawson: *Python Programming for the Absolute Beginner*, 3rd Edition, Course Technology, Boston, 2010

1.12. Number of copies of required reading compared to the number of students currently taking the course

Title	*Number of copies*	*Number of students*
Alan Mutka: Presentations for lectures and practical sessions in the course "Introduction to Programming", Technical College in Bjelovar	0, available *online*	30
Zoran Kalafatić, Antonio Pošćić, Siniša Šegvić, Julijan Šribar: *Python za znatiželjne*, Element, Zagreb, 2016	6	30

1.13. Methods of quality monitoring aimed at ensuring the acquisition of output knowledge, skills and competences

Conducting student surveys and evaluating the data in accordance with the results of the Student Survey Committee and the analyses of exam results in accordance with the data from the ISVU data warehouse.

1.14. Possibility of course instruction in a foreign language

English

GENERAL INFORMATION							
Course title	**Communication Skills**						
Programme of study	Undergraduate professional programme of study in Computer Science						
Course status	Compulsory						
Year	1st						
Semester	1st						
Calculation of ECTS credits	Lectures	Practical sessions		Preparation for practical sessions	Seminar paper	Self-study	Total
		APS	LPS				
	15	30		15		30	90
ECTS credits and forms of instruction	ECTS student workload coefficient				3		
	Contact hours (L+PS+S)				L	PS	S
					15	30	

1. COURSE DESCRIPTION

1.1. Course objectives

1. introducing students to interpersonal communication and the importance of communication skills in engineering.
2. developing skills of identifying and using different communication styles, active listening, perception, speaking and giving feedback.
3. introducing students to the methods and techniques of identifying and resolving complaints and conflicts, techniques and skills of self-presentation, presentation and negotiation.

1.2. Prerequisites

None.

1.3. Expected learning outcomes

Upon completion of the course students will be able to:

1. recognise and describe basic concepts in the field of communication,
2. describe and demonstrate basic features of verbal, nonverbal and paraverbal messages, ,
3. identify and compare basic communication styles and apply an assertive I-message,
4. distinguish between three types of complaints and complaint resolving techniques,
5. demonstrate a balanced feedback,
6. Prepare and deliver a presentation on a given topic,
7. describe and demonstrate the basic elements of the negotiation process,
8. participate in a debate on a given topic.

1.4. Course content

1. Introduction to communication

The notion of communication. Levels of communication phenomena (intrapersonal, interpersonal, group, public and mass communication). Objectives, principles and types of communication.

2. Verbal communication

Factors of efficiency and suitability of verbal communication. Prejudices about communication. Aspects of messages. Criteria of successful verbal communication. Connotative and denotative level of importance of verbal communication.

3. Problems in communication
The communication process. Noisiness, noise, barriers to communication. Types of noise, external and internal barriers to communication.
4. Techniques and skills of active listening
Listening as a physical and mental activity. Types of (not) listening. Principles of active listening.
5. Skills of information gathering
Techniques and skills of asking questions. Types of questions according to the objective of communication.
6. The technique of giving feedback
The concept and purpose of feedback. The five main categories of feedback. A balanced feedback. Receiving and giving praise.
7. Communication styles
Aggressive, submissive-aggressive, passive, assertive communication style. The link between communication styles and the outcome of communication. Notion and meaning of assertiveness. Principles of assertive communication. Structure and effect of I-messages in relation to the YOU-message.
8. Recognizing and resolving complaints
Notion of complaints. Types of complaints. General rules for resolving complaints. Resolving complaints with regard to the type of complaint.
9. Non-verbal and paraverbal communication
Types, characteristics and functions of paraverbal communication. Types and functions of non-verbal messages. Communication skills in business communication.
10. Self-introducing and management of impressions
Skills of impressions management. Five main strategies of self-introducing.
11. Presentation techniques and skills
Preparation and design of presentations. Delivery structure. Verbal and non-verbal elements of delivery in front of an audience. Answering questions.
12. Negotiation
Definition of negotiation and negotiation situations. Characteristics of successful negotiators. Preparing for negotiations.
Negotiating strategies. Tactics and techniques for the initial, central and final phase of negotiations. Unethical techniques / tactics in negotiations.
13. Debate
Notion of debate. Participants in a debate. Debate parts. Debate in the development of critical, logical and creative thinking.

1.5. Forms of instruction	☒ lectures ☐ seminars and workshops ☒ practical sessions ☐ distance learning ☐ field work	☒ individual work ☐ multimedia and web ☐ laboratory sessions ☐ mentorship ☐ other _____
1.6. Comments		

1.7. Students' obligations

2 preliminary exams
3 homework assignments, one of which being the preparation for a presentation
10 written assignments/exercises

1.8. Students' performance monitoring

Class attendance	**0.5**	Class participation	**0.5**	Seminar paper		Experimental work	
Written exam	**1**	Oral exam	**0.5**	Essay		Research	
Project		Continuous assessment	**0.5**	Class report		Practical work	

Portfolio							

1.9. Grading and evaluation of student work in class and in final exam

1. Regular fulfilment of obligations during the semester by which, according to defined criteria and rules, students will collect at least 50 points and meet the requirements for the recognition of the written exam and take the oral exam.

GRADING ELEMENTS	Description	Minimum	Maximum
Class activity	10 written assignments / exercises, each equals 2 points and up to 5 points for presentation delivery	12	25
Homework assignments	3 homework assignments – two written assignments and one presentation	8	15
1st preliminary exam	20 questions	15	30
2nd preliminary exam	20 questions	15	30
	Total	50	100

Students may collect up to 5 additional points during revision in lectures and practical sessions.

Students who achieve a sufficient number of points in both preliminary exams, but does not meet ONE of the other grading elements (activity, homework assignments), may make up for the points by writing a seminar paper. It is possible to make up for 5 points by writing a seminar paper. The topic will be arranged with the course instructor. The seminar paper must be delivered by e-mail not later than 7 days before the exam term.

2. Written and oral exam
Students who do not pass the preliminary exams, but meets other grading requirements, takes the written exam in the part of course material that they have not passed in preliminary exam.
In the oral exam students enter randomly chosen questions from prepared cards into the protocol form. Under the question students enter notes that may be used during the oral exam.

Grading (collected points or the percentage achieved in the written exam):
50 – 62 = sufficient (2)
63 – 75 = good (3)
76 – 88 = very good (4)
89 and more= excellent (5)

1.10. Required reading (at the time of proposing the study programme)

1. Reardon, Kathleen: Interpersonalna komunikacija – Gdje se misli susreću, Alineja, Zagreb, 1998
2. Badrov, Tatjana: Presentations for lectures and practical sessions in the course Communication Skills, http://vtsbj.hr/1-Year-komunikacijske-vjestine-program/komunikacijske-vjestine/ (22.02.2016)

1.11. Further reading (at the time of proposing the study programme)

1. Jandrić, Petar: *Komunikacijske vještine*, http://nastava.tvz.hr/komunikacijske-vjestine/index.php/hr/ , (available as of: 22.02.2016)
2. Fox, Renata: *Poslovna komunikacija*, Hrvatska sveučilišna naklada, Zagreb, 2006

1.12. Number of copies of required reading compared to the number of students currently taking the course

Title	*Number of copies*	*Number of students*
Reardon, Kathleen: Interpersonalna komunikacija – Gdje se misli susreću, Alineja, Zagreb, 1998	6	30

Badrov, Tatjana: Presentations for lectures and practical sessions in the course Communication Skills, http://vtsbj.hr/1-Year-komunikacijske-vjestine-program/komunikacijske-vjestine/	0, available *online*	30

1.13. Methods of quality monitoring aimed at ensuring the acquisition of output knowledge, skills and competences
Conducting student surveys and evaluating the data in accordance with the results of the Student Survey Committee and the analyses of exam results in accordance with the data from the ISVU data warehouse.
1.14. Possibility of course instruction in a foreign language
English

GENERAL INFORMATION							
Course title	**Technical English 1**						
Programme of study	Undergraduate professional programme of study in Computer Science						
Course status	Compulsory						
Year	1st						
Semester	1st						
Calculation of ECTS credits	Lectures	Practical sessions		Preparation for practical sessions	Seminar paper	Self-study	Total
		APS	LPS				
	15	30				15	60
ECTS credits and forms of instruction	ECTS student workload coefficient				2		
	Contact hours (L+PS+S)				L	PS	S
					15	30	

1. COURSE DESCRIPTION

1.1. Course objectives

1. To develop students' ability to use the English language related to specific technical fields.

1.2. Prerequisites

None.

1.3. Expected learning outcomes

Upon completion of the course students will be able to use the English language to:

1. describe technical functions and applications of products and the manner in which products function,
2. categorise technical materials and describe their properties and application,
3. describe the shape and features of components and assemblies as well as manufacturing techniques,
4. demonstrate mastery of simple grammatical structures.

1.4. Course content

1. Technology in Use
Description of technical functions and applications. Emphasising technical advantages. Simplifying technical advantages. Functions and applications of computers. Grammar focus: *Simple Present*, *Present Continuous*, *Simple Past*, *Past Continuous*, future tenses.

2. Technical materials
Categorising technical materials. Naming and describing the properties of technical materials. Comparing technical materials. Describing the application of technical materials. Describing the properties and application of materials used in the manufacturing of computers and electronic and computer assemblies. Grammar focus: adjectives, types of adjectives, comparison of adjectives, modal verbs, types of questions, forming questions.

3.Components and assemblies
Describing the shape and properties of components. Describing manufacturing techniques. Describing fixing and jointing techniques. Describing the position of elements in assemblies. Parts of computers and computer assemblies. Grammar focus: prepositions, relative clauses.

1.5. Forms of instruction	☒ lectures ☐ seminars and workshops ☒ practical sessions ☐ distance learning ☐ field work	☐ individual work ☐ multimedia and web ☐ laboratory sessions ☐ mentorship ☐ other _____
1.6. Comments		

1.7. Students' obligations

2 preliminary exams or the written exam
Oral exam

1.8. Students' performance monitoring

Class attendance		Class participation	**0.5**	Seminar paper		Experimental work	
Written exam	**1**	Oral exam	**0.5**	Essay		Research	
Project		Continuous assessment		Class report		Practical work	
Portfolio							

1.9. Grading and evaluation of student work in class and in final exam

Written exam/Two preliminary exams: 50 points (25 points per each preliminary exam)
Oral exam: 25 points
Class work (practice): 25 points

To pass the written exam, a score of 25 points is required. Passing the written exam (by taking the preliminary exams or the final examination) is a prerequisite for taking the oral exam.
The final grade is based on scores attained in the written exam, oral exam and class work as follows:

50 points - 60 points = sufficient (2)
61 points - 74 points = good (3)
75 points - 87 points = very good (4)
88 points - 100 points = excellent (5)

1.10. Required reading (at the time of proposing the study programme)

1. Ibbotson, Mark: *Cambridge English for Engineering*, Cambridge University Press, Cambridge, 2008

1.11. Further reading (at the time of proposing the study programme)

1. Murphy, Raymond: *English Grammar in Use*, Cambridge University Press, Cambridge, 2004

1.12. Number of copies of required reading compared to the number of students currently taking the course

Title	*Number of copies*	*Number of students*

Ibbotson, Mark: *Cambridge English for Engineering*, Cambridge University Press, Cambridge, 2008	6	30

1.13. Methods of quality monitoring aimed at ensuring the acquisition of output knowledge, skills and competences

Conducting student surveys and evaluating the data in accordance with the results of the Student Survey Committee and the analyses of exam results in accordance with the data from the ISVU data warehouse.

1.14. Possibility of course instruction in a foreign language

English

GENERAL INFORMATION							
Course title	**MATLAB**						
Programme of study	Undergraduate professional programme of study in Computer Science						
Course status	Compulsory						
Year	1st						
Semester	2nd						
Calculation of ECTS credits	Lectures	Practical sessions		Preparation for practical sessions	Seminar paper	Self-study	Total
		APS	LPS				
			30			30	60
ECTS credits and forms of instruction	ECTS student workload coefficient				2		
	Contact hours (L+PS+S)				L	PS	S
					0	30	

1. COURSE DESCRIPTION

1.1. Course objectives

1. Acquiring basic knowledge and skills in working with the mathematical tool *Matlab* and *Simulink*.

1.2. Prerequisites

None.

1.3. Expected learning outcomes

Upon completion of the course students will be able to:

1. use the *Matlab* mathematical tool for matrix calculus, solving systems of equations, derivatives and integrals,
2. calculate and draw graphs of elementary mathematical functions,
3. use symbolic calculation,
4. use *Simulink* to simulate system behavior.

1.4. Course content

1. Introduction to *Matlab*

About *Matlab*. Starting *Matlab*. Organisation of *Matlab* and data structures.

2. Variables

Internal variables. External variables. Vectors. Matrices. Complex numbers. Structures. Erasing variables.

3. Operations in *Matlab*

Arithmetic operators. Relational operators. Logical operators. Conditional statements. Loops.

4. Functions

Elementary mathematical functions. Functions for vectors and matrices. Functions for working with polynomials. M-functions. M-scripts

5. Symbolic mathematical expressions

Basic functions of *Symbolic Toolbox*. Conversion of variables. Simplifying symbolic expressions. Functions for solving equations. Functions for solving equation systems. Functions for derivatives and integrals. Drawing graphs.

6. Graphical functions of *Matlab*

Functions for drawing 2D graphs. Functions for drawing 3D graphs. Functions for drawing surfaces.

7. *Simulink*

Basic actions in *Simulink*. Examples of applying *Simulink* for the simulation of system behaviour

1.5. Forms of instruction	☐ lectures ☐ seminars and workshops ☒ practical sessions ☐ distance learning ☐ field work	☐ individual work ☐ multimedia and web ☒ laboratory sessions ☐ mentorship ☐ other _____
1.6. Comments		

1.7. Students' obligations

1. Continuous assessment.

1.8. Students' performance monitoring

Class attendance	**1**	Class participation		Seminar paper		Experimental work	
Written exam		Oral exam		Essay		Research	
Project		Continuous assessment	**1**	Class report		Practical work	
Portfolio							

1.9. Grading and evaluation of student work in class and in final exam

Continuous knowledge assessment in practical sessions: 10 x 2 points = 20 points.
Two preliminary exams/exam: 80 points
1st preliminary exam: 40 points (a minimum of 15 points is required),
2nd preliminary exam: 40 points (a minimum of 15 points is required).

A maximum of 100 points can be scored in the course. Students pass the written exam if they have achieved a total of 50 points in continuous assessment and preliminary exams/exam. All students who have passed the written exam may take the oral exam. Students who have passed the written exam by passing the preliminary exams are exempt from taking the oral exam, except if they wish to improve their grade.

WRITTEN EXAM GRADING:
[50, 63 > points - sufficient (2)
[63, 75 > points - good (3)
[75, 87 > points - very good (4)
[87, 100] points - excellent (5)

1.10. Required reading (at the time of proposing the study programme)

1. Ban, Željko; Matuško, Jadranko; Petrović, Ivan: *Primjena programskog sustava MATLAB za rješavanje tehničkih problema*, Graphis, Zagreb, 2010

1.11. Further reading (at the time of proposing the study programme)

1. MathWorks: *MATLAB ProductHelp*,TheMathWorksInc., Natick, 2013

1.12. Number of copies of required reading compared to the number of students currently taking the course

Title	*Number of copies*	*Number of students*
Ban, Željko; Matuško, Jadranko; Petrović, Ivan: *Primjena programskog sustava MATLAB za rješavanje tehničkih problema*, Graphis, Zagreb, 2010	6	30

1.13. Methods of quality monitoring aimed at ensuring the acquisition of output knowledge, skills and competences

Conducting student surveys and evaluating the data in accordance with the results of the Student Survey Committee and the analyses of exam results in accordance with the data from the ISVU data warehouse.

1.14. Possibility of course instruction in a foreign language

English

GENERAL INFORMATION							
Course title	**Introduction to Computer Networks**						
Programme of study	Undergraduate professional programme of study in Computer Science						
Course status	Compulsory						
Year	1st						
Semester	2nd						
Calculation of ECTS credits	Lectures	Practical sessions		Preparation for practical sessions	Seminar paper	Self-study	Total
		APS	LPS				
	30	0	15	15	0	90	150
ECTS credits and forms of instruction	ECTS student workload coefficient				5		
	Contact hours (L+PS+S)				L	PS	S
					30	15	

1. COURSE DESCRIPTION

1.1. Course objectives

1. Understanding the functioning of computer networks and standard communication protocols.
2. Acquiring the basic knowledge of connecting computer networks.

1.2. Prerequisites

None.

1.3. Expected learning outcomes

Upon completion of the course students will be able to:

1. define the basic principle of how computer communication systems and computer networks operate,
2. describe the operation of a standard TCP/IP protocol,
3. describe the procedure of connecting several computers into a network,
4. recognise the purpose of various network devices.

1.4. Course content

1. **Introduction to computer networks**
   History. Organisation of computer networks. Parameters and classification of networks according to various criteria. Network standards.
2. **Network architectures: OSI model and TCP/IP model**
   Architectures and concept of computer networks. ISO/OSI standard. TCP/IP architectures.
3. **Physical layer**
   Transmission media (wired and wireless). Signal modulation. Local area networks (LAN – Ethernet / IEEE 802.3). MAC address. Structure of the Ethernet frame. Structured cabling. Connecting local networks.
4. **Network layer**
   Network layer protocols. IP protocol (IPv4). IP address. Addressing in the network. Structure of the IP package. IP fragmentation. Network routing. IP protocol IPv6. Comparison between IPV4 and IPV6.
5. **Transmission layer**
   TCP protocol. Setting up and ending TCP connection. UDP protocol. Flow management.
6. **Application layer**
   Display of the application layer protocol: HTTP, FTP, Telnet, SMTP. DNS system.

7. Connecting a local area network to the Internet. Access technologies (ADSL). Related protocols (NAT, DHCP)

1.5. Forms of instruction	☒ lectures ☐ seminars and workshops ☒ practical sessions ☐ distance learning ☐ field work	☐ individual work ☐ multimedia and web ☒ laboratory sessions ☐ mentorship ☐ other _____
1.6. Comments		

1.7. Students' obligations

2 preliminary exams
2 homework assignments
6 laboratory sessions

1.8. Students' performance monitoring

Class attendance	**0.5**	Class participation		Seminar paper		Experimental work	0
Written exam	**2.5**	Oral exam	**1**	Essay		Research	
Project		Continuous assessment	**1**	Class report		Practical work	0
Portfolio							

1.9. Grading and evaluation of student work in class and in final exam

Pop quizzes/Homework assignments: 5 x 2 points = 10 points
Activity in laboratory sessions: 30 points
Two preliminary exams: 60 points (a minimum of 20 points is required),
1st preliminary exam: 30 points (a minimum of 10 points is required),
2nd preliminary exam: 30 points (a minimum of 10 points is required).

A maximum of 100 points can be scored in the course. Students pass the written exam if they have achieved a total of 50 points doing their homework assignments, laboratory practice and preliminary exams. Students who have passed the written exam by passing the preliminary exams are exempt from taking the oral exam, except if they wish to improve their grade.

Grading and evaluation of students' work upon completion of the course:
Written exam: 100 points
Oral exam
Students pass the written exam if they have scored 50 points. After they have passed the written exam, students take the oral exam.

WRITTEN EXAM GRADING:
[50, 63> points – sufficient (2)
[63, 75> points – good (3)
[75, 87> points – very good (4)
[87, 100] points – excellent (5)

1.10. Required reading (at the time of proposing the study programme)

- A. Bažant et al.: *Osnove arhitekture mreža*, Element, Zagreb, 2014
- Robert Herčeki: Presentations for lectures and practical sessions in the course "Introduction to Computer Networks", Technical College in Bjelovar

1.11. Further reading (at the time of proposing the study programme)

- A. S. Tanenbaum, D. J. Wetherall: *Computer Networks*, 5th Ed., Prentice Hall, 2011

1.12. Number of copies of required reading compared to the number of students currently taking the course

Title	*Number of copies*	*Number of students*
A. Bažant et al.: *Osnove arhitekture mreža*, Element, Zagreb, 2014	6	30
Robert Herčeki: Presentations for lectures and practical sessions in the course "Introduction to Computer Networks", Technical College in Bjelovar	0, available *online*	30

1.13. Methods of quality monitoring aimed at ensuring the acquisition of output knowledge, skills and competences

Conducting student surveys and evaluating the data in accordance with the results of the Student Survey Committee and the analyses of exam results in accordance with the data from the ISVU data warehouse.

1.14. Possibility of course instruction in a foreign language

English

GENERAL INFORMATION							
Course title	**Introduction to Linux**						
Programme of study	Undergraduate professional programme of study in Computer Science						
Course status	Compulsory						
Year	1st						
Semester	2nd						
Calculation of ECTS credits	Lectures	Practical sessions		Preparation for practical sessions	Seminar paper	Self-study	Total
		APS	LPS				
	30	0	30	30	0	60	150
ECTS credits and forms of instruction	ECTS student workload coefficient				5		
	Contact hours (L+PS+S)				L	PS	S
					30	30	

1. COURSE DESCRIPTION

1.1. Course objectives

1. Understanding the fundamentals of working in the operating system *Linux*.
2. Understanding the advantages of an open code system.

1.2. Prerequisites

None.

1.3. Expected learning outcomes

Upon completion of the course students will be able to:
1. manage in the operating system *Linux*,
2. control the system *Linux* from the command line shell,
3. understand the difference between *Unix*/*Linux* basic components,
4. describe the procedure of connecting several computers into a network.

1.4. Course content

1. **Introduction to *Linux***
   History of *Unix*. Advantages of the open code system *Linux*.
2. **Getting familiarised with own computer in the world of *Linux***
   Overview of computer components. Configuration of computer settings.
3. **Getting familiarised with *Unix*/*Linux* components**
   Kernel. System libraries. Shell. *Linux* commands. File System. Path. Authorisation. Processes. Meta signs. Rerouting.
4. **Working in the operating system *Linux***
   Basic *Linux* commands. Working with file content. Searching for files. *Shell* scripts. Discs and partitions. RAM. Administering the *Linux* system. Archiving and compressing / decompressing data.
5. **Network systems**
   Network terms. Communication protocols. Network operation. Network configuration in *Linux*. Basic network tools. Network services. Firewall.

1.5. Forms of instruction	☒ lectures ☐ seminars and workshops ☒ practical sessions ☐ distance learning ☐ field work	☐ individual work ☐ multimedia and web ☒ laboratory sessions ☐ mentorship ☐ other _____
1.6. Comments		

1.7. Students' obligations

2 preliminary exams
2 homework assignments
15 laboratory session

1.8. Students' performance monitoring

Class attendance	**0.5**	Class participation		Seminar paper		Experimental work	0
Written exam	**2**	Oral exam	**1.5**	Essay		Research	
Project		Continuous assessment	**1**	Class report		Practical work	0
Portfolio							

1.9. Grading and evaluation of student work in class and in final exam

Homework assignments: 5 x 2 points = 10 points
Work in laboratory sessions: 30 points
Two preliminary exams/exam: 60 points (a minimum of 20 points is required),
1st preliminary exam: 30 points (a minimum of 10 points is required),
2nd preliminary exam: 30 points (a minimum of 10 points is required).

A maximum of 100 points can be scored in the course. Students pass the written exam if they have achieved a total of 50 points in homework assignments, laboratory sessions and preliminary exams. Students who have passed the written exam by passing the preliminary exams are exempt from taking the oral exam, except if they wish to improve their grade.

Grading and evaluation of students' work upon completion of the course:
Written exam: 100 points
Oral exam

Students pass the written exam if they have scored 50 points. After they have passed the written exam, students take the oral exam.

WRITTEN EXAM GRADING:
[50, 63 > points - sufficient (2)
[63, 75 > points - good (3)
[75, 87 > points - very good (4)
[87, 100] points - excellent (5)

1.10. Required reading (at the time of proposing the study programme)

- Hrvoje Horvat: *Uvod u Linux*, Open Source Osijek, online edition (https://www.opensource-osijek.org/dokuwiki/wiki:knjige:uvod_u_linux)
- Dario Vidić: Presentations for lectures and practical sessions in the course "Introduction to Linux", Technical College in Bjelovar

1.11. Further reading (at the time of proposing the study programme)

- Brian Ward: *Kako radi Linux*, Dobar plan, Zagreb, 2016

1.12. Number of copies of required reading compared to the number of students currently taking the course

Title	*Number of copies*	*Number of students*
Hrvoje Horvat: *Uvod u Linux*, Open Source Osijek, online edition (https://www.opensource-osijek.org/dokuwiki/wiki:knjige:uvod_u_linux)	0, available *online*	30
Dario Vidić: Presentations for lectures and practical sessions in the course "Introduction to Linux", Technical College in Bjelovar	0, available *online*	30

1.13. Methods of quality monitoring aimed at ensuring the acquisition of output knowledge, skills and competences

Conducting student surveys and evaluating the data in accordance with the results of the Student Survey Committee and the analyses of exam results in accordance with the data from the ISVU data warehouse.

1.14. Possibility of course instruction in a foreign language

English

GENERAL INFORMATION							
Course title	**Programming in C**						
Programme of study	Undergraduate professional programme of study in Computer Science						
Course status	Compulsory						
Year	1st						
Semester	2nd						
Calculation of ECTS credits	Lectures	Practical sessions		Preparation for practical sessions	Seminar paper	Self-study	Total
		APS	LPS				
	30	0	30	30	0	90	180
ECTS credits and forms of instruction	ECTS student workload coefficient				6		
	Contact hours (L+PS+S)				L	PS	S
					30	30	

1. COURSE DESCRIPTION

1.1. Course objectives

1. Acquiring basic knowledge in fundamentals of programming in C.
2. Learning how to use a development environment for developing a program in C.
3. Learning how to create simple programs.

1.2. Prerequisites

None.

1.3. Expected learning outcomes

Upon completion of the course students will be able to:

1. program in C,
2. create simple programs,
3. use the development environment for developing a program in C,
4. use defined functions and create own functions

1.4. Course content

1. Programming languages and programming
History of programming languages. Potential application of programming. Code development.
2. Programming language C
Writing a code in C. Development environment *Code::Blocks*. Pre-processor commands. Input-output flow commands.
3. Types of data and arithmetic operators
Types of data. Declaration of variables. Arithmetic operations. Assignment operators.
4. Logical types of data and operators
Logical types of data. Comparison operators.
5. Bitwise operators

6. Functions
Declaration of a function. Definition of a function. Standard headers. Header. Calling a function. Opensource functions.
7. Command blocks and conditional code execution
Code execution order. Command blocks. If command block. Switch-case command block.
8. Programming loops
For loop. While loop. Do-while loop. Commands break and continue.
9. Fields
One-dimensional fields. Bubblesort. Two-dimensional fields.
10. Strings
Signs. Strings. Functions for working with strings.
11. Files
Loading and writing binary and textual files. Entering complex data into a file.
12. Pointers
Pointers for working with variables, functions and fields.

1.5. Forms of instruction	☒ lectures ☐ seminars and workshops ☒ practical sessions ☐ distance learning ☐ field work	☐ individual work ☐ multimedia and web ☒ laboratory sessions ☐ mentorship ☐ other _____
1.6. Comments		

1.7. Students' obligations

2 preliminary exams
4 homework assignments
4 short, unannounced tests
15 laboratory sessions

1.8. Students' performance monitoring

Class attendance	**0,25**	Class participation		Seminar paper		Experimental work	**1**
Written exam	**0.75**	Oral exam	**0.5**	Essay		Research	
Project		Continuous assessment	**0.5**	Class report		Practical work	
Portfolio							

1.9. Grading and evaluation of student work in class and in final exam

Homework assignments: 5 x 2 points = 10 points
Work in laboratory sessions: 30 points
Two preliminary exams/exam: 60 points (a minimum of 20 points is required),
1st preliminary exam: 30 points (a minimum of 10 points is required),
2nd preliminary exam: 30 points (a minimum of 10 points is required).

A maximum of 100 points can be scored in the course. Students pass the written exam if they have achieved a total of 50 points in homework assignments, laboratory sessions and preliminary exams. Students who have passed the written exam by passing the preliminary exams are exempt from taking the oral exam, except if they wish to improve their grade.

Grading and evaluation of student work after the course:
Written exam: 100 points
Oral exam

Students pass the written exam if they have scored 50 points. After they have passed the written exam, students take the oral exam.
.

WRITTEN EXAM GRADING:
[50, 63 > points - sufficient (2)
[63, 75 > points - good (3)
[75, 87 > points - very good (4)
[87, 100] points - excellent (5)

1.10. Required reading (at the time of proposing the study programme)

1. Domagoj Kusalić: *Napredno programiranje i algoritmi u C-u i C++-u*, 5th edition, Element, Zagreb, 2014
2. Alan Mutka, Zoran Vrhovski: Presentations for lectures and practical sessions in "Programming in C", Technical College in Bjelovar, Bjelovar, 2016, http://vtsbj.hr/1-Year-osnove-programiranja-pred-vjezb/ (available as of: 21.2.2016)

1.11. Further reading (at the time of proposing the study programme)

1. D. M. Ritchie, B. W. Kernighan (Translation: Ante Denić): *Programski jezik C*, secord edition (https://www.scribd.com/doc/47734390/Programski-jezik-C)
2. J. Šribar, B. Motik*: Demistificirani C++*, 3rd edition, Element, Zagreb, 2010
3. *Learn C programming*, http://www.tutorialspoint.com/cprogramming/ (available as of: 21.2.2016)
4. *C Tutorials*, http://www.codingunit.com/category/c-tutorials (available as of: 21.2.2016)

1.12. Number of copies of required reading compared to the number of students currently taking the course

Title	*Number of copies*	*Number of students*
Domagoj Kusalić: *Napredno programiranje i algoritmi u C-u i C++-u*, 5th edition, Element, Zagreb, 2014	6	30
Alan Mutka, Zoran Vrhovski: Presentations for lectures and practical sessions in "Programming in C", Technical College in Bjelovar, Bjelovar, 2016	0, available *online*	30

1.13. Methods of quality monitoring aimed at ensuring the acquisition of output knowledge, skills and competences

Conducting student surveys and evaluating the data in accordance with the results of the Student Survey Committee and the analyses of exam results in accordance with the data from the ISVU data warehouse.

1.14. Possibility of course instruction in a foreign language

English

GENERAL INFORMATION							
Course title	**Technical English 2**						
Programme of study	Undergraduate professional programme of study in Computer Science						
Course status	Compulsory						
Year	1st						
Semester	2nd						
Calculation of ECTS credits	Lectures	Practical sessions		Preparation for practical sessions	Seminar paper	Self-study	Total
		APS	LPS				
	15	30				15	60
ECTS credits and forms of instruction	ECTS student workload coefficient				2		
	Contact hours (L+PS+S)				L	PS	S
					15	30	

1. COURSE DESCRIPTION
1.1. Course objectives
1. To develop students' ability to use the English language related to specific technical fields.
1.2. Prerequisites
Previous enrolment in Technical English 1.
1.3. Expected learning outcomes
Upon completion of the course students will be able to use the English language to: 1. describe the position of components in assemblies, technical drawings, design procedure and dimensions, precision and tolerances in the preparation of technical documentation, 2. describe technical problems and malfunctions as well as their causes and possible solutions, 3. discuss about technical requirements and describe project feasibility, improvements and redesigns, 4. demonstrate mastery of simple grammatical structures.
1.4. Course content
1. Technical drawings Describing the procedure of making drawings. Dimensioning, precision, tolerances. Design process. Software tools. Grammar focus: numbers, conjunctions, pronouns, determiners. **2.Repair and maintenance** Describing technical malfunctions. Assessment of faults. Describing the causes of malfunctions. Discussing repair and maintenance. Computer maintenance. Duties of a system engineer. Grammar focus: *Present Perfect*, comparison between *Present Perfect* and *Simple Past*. **3. Technical development** Describing technical requirements. Suggesting ideas and solutions. Feasibility study. Describing improvements and redesigns. Development of software tools in business environment. Grammar focus: *Past Perfect*. Comparison between *Past Perfect* and *Simple Past*.

1.5. Forms of instruction	☒ lectures ☐ seminars and workshops ☒ practical sessions	☐ individual work ☐ multimedia and web ☐ laboratory sessions

	☐ distance learning ☐ field work	☐ mentorship ☐ other _____
1.6. Comments		
1.7. Students' obligations		
2 preliminary exams or the written exam Oral exam		
1.8. Students' performance monitoring		

Class attendance		Class participation	**0.5**	Seminar paper		Experimental work	
Written exam	**1**	Oral exam	**0.5**	Essay		Research	
Project		Continuous assessment		Class report		Practical work	
Portfolio							

1.9. Grading and evaluation of student work in class and in final exam

Written exam/Two preliminary exams: 50 points (25 points per each preliminary exam)
Oral exam: 25 points
Class work (practice): 25 points

To pass the written exam, a score of 25 points is required. Passing the written exam (by taking the preliminary exams or the final examination) is a prerequisite for taking the oral exam.
The final grade is based on scores attained in the written exam, oral exam and class work as follows:

50 points - 60 points = sufficient (2)
61 points - 74 points = good (3)
75 points - 87 points = very good (4)
88 points - 100 points = excellent (5)

1.10. Required reading (at the time of proposing the study programme)

1. Ibbotson, Mark: *Cambridge English for Engineering*, Cambridge University Press, Cambridge, 2008

1.11. Further reading (at the time of proposing the study programme)

1. Murphy, Raymond: *English Grammar in Use*, Cambridge University Press, Cambridge, 2004

1.12. Number of copies of required reading compared to the number of students currently taking the course

Title	*Number of copies*	*Number of students*
Ibbotson, Mark: *Cambridge English for Engineering*, Cambridge University Press, Cambridge, 2008	6	30

1.13. Methods of quality monitoring aimed at ensuring the acquisition of output knowledge, skills and competences		
Conducting student surveys and evaluating the data in accordance with the results of the Student Survey Committee and the analyses of exam results in accordance with the data from the ISVU data warehouse.		
1.14. Possibility of course instruction in a foreign language		
English		

GENERAL INFORMATION							
Course title	**Digital Techniques**						
Programme of study	Undergraduate professional programme of study in Computer Science						
Course status	Compulsory						
Year	2nd						
Semester	3rd						
Calculation of ECTS credits	Lectures	Practical sessions		Preparation for practical sessions	Seminar paper	Self-study	Total
		APS	LPS				
	30	16	14	30		90	180
ECTS credits and forms of instruction	ECTS student workload coefficient				6		
	Contact hours (L+PS+S)				L	PS	S
					30	30	

1. COURSE DESCRIPTION
1.1. Course objectives
1. To acquire basic knowledge of digital techniques. 2. To understand the operation of digital circuits.
1.2. Prerequisites
None.
1.3. Expected learning outcomes
Upon completion of the course students will be able to: 1. perform conversions between different number systems, 2. minimise and realise complex logical functions using logic circuits, 3. draw and explain basic logic circuits in semiconductor technology, 4. draw and explain complex combinational modules, 5. draw and explain types of bistables, registers and counters, 6. draw and explain operation of AD and DA conversion circuits.
1.4. Course content
1. Number systems and codes Number systems (decimal, binary, hexadecimal etc.). Conversion of numbers between number systems. Operations with binary numbers. Characteristic binary codes. Binary encoding. **2. Logic circuits** Propositional logic. The basic principles of Boolean algebra. AND, OR, NOT, NAND, NOR gates. Complex logical operations. Minterms and maxterms. Methods of minimisation (K-map, the Quine-McCluskey method). **3. Design of logic circuits in semiconductor technology** TTL technology. CMOS technology. **4. Complex combinational modules**

Adders. Digital comparator. Parity circuit. Encoder and decoder. Multiplexer and demultiplexer.

5. Bistables

Operation and types of bistables. Bistable design using logic integrated circuits.

6. Registers and counters

Design and implementation of registers. Design of counters. Asynchronous and synchronous counters. Decade counters. Implementation of counters. Integrated counters.

7. D/A and A/D conversion

1.5. Forms of instruction	☒ lectures ☐ seminars and workshops ☒ practical sessions ☐ distance learning ☐ field work	☐ individual work ☐ multimedia and web ☒ laboratory sessions ☐ mentorship ☐ other _____
1.6. Comments		

1.7. Students' obligations

2 preliminary exams
5 homework assignments

1.8. Students' performance monitoring

Class attendance	**0.5**	Class participation		Seminar paper		Experimental work	**1.5**
Written exam	**1.5**	Oral exam	**1.5**	Essay		Research	
Project		Continuous assessment	**1**	Class report		Practical work	
Portfolio							

1.9. Grading and evaluation of student work in class and in final exam

Grading and evaluation of students' work in the course:
Homework assignments: 5 x 2 points = 10 points
Student performance in laboratory sessions: 30 points
Two preliminary exams: 60 points (a minimum of 20 points is required),
1st preliminary exam: 30 points (a minimum of 10 points is required),
2nd preliminary exam: 30 points (a minimum of 10 points is required).

A maximum of 100 points can be scored in the course. Students pass the written exam if they have achieved a total of 50 points doing their homework assignments, laboratory practice and preliminary exams. Students who have passed the written exam by passing the preliminary exams are exempt from taking the oral exam, except if they wish to improve their grade.

Grading and evaluation of students' work upon completion of the course:
Written exam: 100 points
Oral exam

Students pass the written exam if they have scored 50 points. After they have passed the written exam, students take the oral exam.

WRITTEN EXAM GRADING:
[50, 63> points - sufficient (2)
[63, 75> points - good (3)

[75, 87> points - very good (4)
[87, 100] points - excellent (5)

1.10. Required reading (at the time of proposing the study programme)

1. Vrhovski, Zoran; Šumiga, Ivan: *Digitalna tehnika* – problem book, Technical College in Bjelovar, Bjelovar, 2015

1.11. Further reading (at the time of proposing the study programme)

1. Peruško, Uroš: *Digitalna elektronika*, Školska knjiga, Zagreb, 1996
2. Čupić, Marko: *Digitalna elektronika i digitalna logika*, problem book, Kigen d.o.o., Zagreb, 2006

1.12. Number of copies of required reading compared to the number of students currently taking the course

Title	*Number of copies*	*Number of students*
Vrhovski, Zoran; Šumiga, Ivan: *Digitalna tehnika* – problem book, Technical College in Bjelovar, Bjelovar, 2015	6	30

1.13. Methods of quality monitoring aimed at ensuring the acquisition of output knowledge, skills and competences

Conducting student surveys and evaluating the data in accordance with the results of the Student Survey Committee and the analyses of exam results in accordance with the data from the ISVU data warehouse.

1.14. Possibility of course instruction in a foreign language

English

GENERAL INFORMATION							
Course title	**Signals and Systems**						
Programme of study	Undergraduate professional programme of study in Computer Science						
Course status	Compulsory						
Year	2nd						
Semester	3rd						
Calculation of ECTS credits	Lectures	Practical sessions		Preparation for practical sessions	Seminar paper	Self-study	Total
		APS	LPS				
	30	24	6	30		60	150
ECTS credits and forms of instruction	ECTS student workload coefficient				5		
	Contact hours (L+PS+S)				L	PS	S
					30	30	

1. COURSE DESCRIPTION

1.1. Course objectives

1. Familiarising students with types of signals and systems.
2. Learning to solve differential and difference equations.
3. Mastering the Laplace and Z transforms as aids to solving differential and difference equations.
4. Understanding Fourier series and the Fourier transform.
5. Mastering the use of *Matlab* and *Simulink* for analysis of signals and systems.

1.2. Prerequisites

None.

1.3. Expected learning outcomes

Upon completion of the course students will be able to:

1. identify types of signals and systems,
2. solve differential equations using the direct method and the Laplace transform,
3. solve difference equations using the direct method and Z-transform,
4. calculate the Fourier series of a periodic signal,
5. calculate the Fourier transform of signals,
6. use *Matlab* and *Simulink* for analysis of signals and systems.

1.4. Course content

1. Fundamental concepts

Types of signals. Types of systems. The concept and basic features of signals and systems. Examples of signals and systems.

2. Linear differential equations

Describing a system using the linear differential equation. Solving homogeneous differential equations. Solving nonhomogeneous differential equations. Block diagram representation of a system. The convolution integral. The impulse response of a system.

3. The Laplace transform
Fundamental features of the direct and the inverse Laplace transforms. Applications of the Laplace transform. The transfer function.
4. Fourier series and the Fourier transform
Fourier series: trigonometric and complex. Harmonics. The Fourier transform.
5. Linear difference equations
Describing a system using the difference equation. Solving homogeneous differential equations. Solving nonhomogeneous differential equations.
6. The Z-transform
Fundamental features of the direct and the inverse Z-transforms. Applications of the Z-transform. The transfer function.

1.5. Forms of instruction	☒ lectures ☐ seminars and workshops ☒ practical sessions ☐ distance learning ☐ field work	☒ individual work ☒ multimedia and web ☒ laboratory sessions ☐ mentorship ☐ other _____
1.6. Comments		

1.7. Students' obligations

2 preliminary exams
5 homework assignments
project work

1.8. Students' performance monitoring

Class attendance	**0.3**	Class participation		Seminar paper		Experimental work	**0.2**
Written exam	**1.25**	Oral exam	**1.25**	Essay		Research	
Project		Continuous assessment	**1**	Class report		Practical work	**1**
Portfolio							

1.9. Grading and evaluation of student work in class and in final exam

Grading and evaluation of students' work in the course:
Homework assignments: 5 x 2 points = 10 points
Project work: 30 points
Two preliminary exams: 60 points (a minimum of 20 points is required),
1st preliminary exam: 30 points (a minimum of 10 points is required),
2nd preliminary exam: 30 points (a minimum of 10 points is required).

A maximum of 100 points can be scored in the course. Students pass the written exam if they have achieved a total of 50 points doing their homework assignments, project work and preliminary exams. Students who have passed the written exam by passing the preliminary exams are exempt from taking the oral exam, except if they wish to improve their grade.

Grading and evaluation of student work after the course:
Written exam: 100 points
Oral exam

Students pass the written exam if they have scored 50 points. After they have passed the written exam, students take the oral exam.

WRITTEN EXAM GRADING:
[50, 63> points - sufficient (2)
[63, 75> points - good (3)
[75, 87> points - very good (4)
[87, 100] points - excellent (5)

1.10. Required reading (at the time of proposing the study programme)

1. Vrhovski, Zoran; Purković, Dalibor: *Signali i sustavi* – problem book, Technical College in Bjelovar, Bjelovar, 2016
2. Vrhovski, Zoran: Presentations for lectures in Signals and Systems, Technical College in Bjelovar, Bjelovar, 2016., http://vtsbj.hr/signali-i-sustavi-predavanja-vjezbe/ (available: 21 February 2016)

1.11. Further reading (at the time of proposing the study programme)

1. Vrankić, Miroslav: *Signali i sustavi* – Problem book, Graphis, Zagreb, 2007

1.12. Number of copies of required reading compared to the number of students currently taking the course

Title	*Number of copies*	*Number of students*
Vrhovski, Zoran; Purković, Dalibor: *Signali i sustavi* – problem book, Technical College in Bjelovar, Bjelovar, 2016	6	30
Vrhovski, Zoran: Presentations for lectures in Signals and Systems, Technical College in Bjelovar, Bjelovar, 2016	0, available online	30

1.13. Methods of quality monitoring aimed at ensuring the acquisition of output knowledge, skills and competences

Conducting student surveys and evaluating the data in accordance with the results of the Student Survey Committee and the analyses of exam results in accordance with the data from the ISVU data warehouse.

1.14. Possibility of course instruction in a foreign language

English

GENERAL INFORMATION							
Course title	**Object-Oriented Programming**						
Programme of study	Undergraduate professional programme of study in Computer Science						
Course status	Compulsory						
Year	2nd						
Semester	3rd						
Calculation of ECTS credits	Lectures	Practical sessions		Preparation for practical sessions	Seminar paper	Self-study	Total
		APS	LPS				
	30	0	30	30	0	90	180
ECTS credits and forms of instruction	ECTS student workload coefficient				6		
	Contact hours (L+PS+S)				L	PS	S
					30	30	

1. COURSE DESCRIPTION

1.1. Course objectives

1. To acquire fundamental knowledge in C++ programming.
2. To master the use of object-oriented approach in programming.
3. To learn how to programme simple programmes using the object-oriented programming approach.

1.2. Prerequisites

Passed exam in **Introduction to Programming**.

1.3. Expected learning outcomes

Upon completion of the course students will be able to:

1. programme using the syntax of the C++ programming language,
2. apply the object-oriented programming methods in the context of simple programmes,
3. use the object-oriented programming approach with the aim of an improved realisation of the target application.

1.4. Course content

1. **Object-oriented programming languages**
   An overview of programming techniques. An introduction to object-oriented programming.
2. **Elements of C++ programs**
   Variables and constants. Expressions and statements. Functions. Loops.
3. **Pointers and references**
   Types of pointers. Dangling pointers. References. Memory management using the *new* and *delete* operators.
4. **The basics of classes**
   Class. Class members. Accessing class members. Visibility of data in a class. *Interface* and implementation. Friend class. The *this* keyword. Constructors and destructors. The *Const* functions. *Volatile*. Static class members. Pointers and classes.
5. **Member functions and operators**
6. **Arrays**
   Arrays and objects. Arrays of pointers. Pointer to an array. Deleting arrays.

7. Inheritance
Hierarchical inheritance. The constructor and destructor. Hiding of functions in base class. Virtual functions.

8. Polymorphism
Virtual inheritance. Abstract data types. Advanced inheritance.

9. Input and output streams
The *CIN* and *COUT* objects. Member functions of objects. File input and output.

1.5. Forms of instruction	☒ lectures ☐ seminars and workshops ☒ practical sessions ☐ distance learning ☐ field work	☐ individual work ☐ multimedia and web ☒ laboratory sessions ☐ mentorship ☐ other _____
1.6. Comments		

1.7. Students' obligations

2 preliminary exams
2 homework assignments
15 laboratory sessions

1.8. Students' performance monitoring

Class attendance	**0.5**	Class participation		Seminar paper		Experimental work	0
Written exam	**2.5**	Oral exam	**1**	Essay		Research	
Project		Continuous assessment	**0.5**	Class report		Practical work	**1.5**
Portfolio							

1.9. Grading and evaluation of student work in class and in final exam

Homework assignments: 5 x 2 points = 10 points
Work in laboratory sessions: 30 points
Two preliminary exams/exam: 60 points (a minimum of 20 points is required),
1st preliminary exam: 30 points (a minimum of 10 points is required),
2nd preliminary exam: 30 points (a minimum of 10 points is required).

A maximum of 100 points can be scored in the course. Students pass the written exam if they have achieved a total of 50 points doing their homework assignments, laboratory practice and preliminary exams.
Students who have passed the written exam by passing the preliminary exams are exempt from taking the oral exam, except if they wish to improve their grade.

Grading and evaluation of student work after the course:
Written exam: 100 points
Oral exam

Students pass the written exam if they have scored 50 points. After they have passed the written exam, students take the oral exam.

WRITTEN EXAM GRADING:
[50, 63 > points - sufficient (2)
[63, 75 > points - good (3)
[75, 87 > points - very good (4)

[87, 100] points - excellent (5)

1.10. Required reading (at the time of proposing the study programme)

- Julijan Šribar, Boris Motik: *Demistificirani C++*, 4th edition, Element, Zagreb, 2014
- Mihael Kukec: Presentations for lectures and practical sessions in Object-Oriented Programming, Technical College in Bjelovar

1.11. Further reading (at the time of proposing the study programme)

- Domagoj Kusalić: *Napredno programiranje i algoritmi u C-u i C++-u*, 5th edition, Element, Zagreb, 2014

1.12. Number of copies of required reading compared to the number of students currently taking the course

Title	*Number of copies*	*Number of students*
Julijan Šribar, Boris Motik: *Demistificirani c++,* 4th edition, Element, Zagreb, 2014	6	30
Mihael Kukec: Presentations for lectures and practical sessions in Object-Oriented Programming, Technical College in Bjelovar	0, available online	30

1.13. Methods of quality monitoring aimed at ensuring the acquisition of output knowledge, skills and competences

Conducting student surveys and evaluating the data in accordance with the results of the Student Survey Committee and the analyses of exam results in accordance with the data from the ISVU data warehouse.

1.14. Possibility of course instruction in a foreign language

English

GENERAL INFORMATION							
Course title	**Data Structures and Algorithms**						
Programme of study	Undergraduate professional programme of study in Computer Science						
Course status	Compulsory						
Year	2nd						
Semester	3rd						
Calculation of ECTS credits	Lectures	Practical sessions		Preparation for practical sessions	Seminar paper	Self-study	Total
		APS	LPS				
	30	0	30	15	0	75	150
ECTS credits and forms of instruction	ECTS student workload coefficient				5		
	Contact hours (L+PS+S)				L	PS	S
					30	30	

1. COURSE DESCRIPTION		
1.1. Course objectives		
1. To understand abstract data types and data structures, 2. To understand the fundamental techniques of algorithm design and analysis.		
1.2. Prerequisites		
Passed exam in **Introduction to Programming**.		
1.3. Expected learning outcomes		
Upon completion of the course students will be able to: 1. use the explained techniques of programming simple programs, 2. use lists, stacks and trees in programming, 3. programme simple searching algorithms, 4. use data searching and manipulation algorithms.		
1.4. Course content		
1. Introduction Fundamental concepts. Data structures. Writing and analysing algorithms. **2. Overview of various abstract data types** List. Stack. Queue. Binary tree. K-tree. Set. Dictionary. Priority queue. Copying. **3. Data structures** Linked list. *Hash* table. Binary search tree. Heap. Binary tree. K-tree. **4. Algorithms for performing basic operations on data structures** Data input and output, searching, printing content. **5. Implementation of the aforementioned structures in complex algorithms** Data array sorting and compression, diverse recursive procedures. **6. General algorithm design techniques:** „divide and conquer“, dynamic programming, the „greedy method“, backtracking.		
1.5. Forms of instruction	☒ lectures ☐ seminars and workshops ☒ practical sessions ☐ distance learning	☐ individual work ☐ multimedia and web ☒ laboratory sessions ☐ mentorship

	☐ field work	☐ other _____
1.6. Comments		

1.7. Students' obligations

2 preliminary exams
2 homework assignments
15 laboratory sessions

1.8. Students' performance monitoring

Class attendance	**0.5**	Class participation		Seminar paper		Experimental work	0
Written exam	**2**	Oral exam	**1**	Essay		Research	
Project		Continuous assessment	**0.5**	Class report		Practical work	**1**
Portfolio							

1.9. Grading and evaluation of student work in class and in final exam

Homework assignments: 5 x 2 points = 10 points
Work in laboratory sessions: 30 points
Two preliminary exams/exam: 60 points (a minimum of 20 points is required),
1st preliminary exam: 30 points (a minimum of 10 points is required),
2nd preliminary exam: 30 points (a minimum of 10 points is required).

A maximum of 100 points can be scored in the course. Students pass the written exam if they have achieved a total of 50 points doing their homework assignments, laboratory practice and preliminary exams. Students who have passed the written exam by passing the preliminary exams are exempt from taking the oral exam, except if they wish to improve their grade.

Grading and evaluation of student work after the course:
Written exam: 100 points
Oral exam

Students pass the written exam if they have scored 50 points. After they have passed the written exam, students take the oral exam.

WRITTEN EXAM GRADING:
[50, 63 > points - sufficient (2)
[63, 75 > points - good (3)
[75, 87 > points - very good (4)
[87, 100] points - excellent (5)

1.10. Required reading (at the time of proposing the study programme)

- Domagoj Kusalić: *Napredno programiranje i algoritmi u C-u i C++-u*, 5th edition, Element, Zagreb, 2014
- Dario Vidić: Presentations for lectures and practical sessions in Data Structures and Algorithms, Technical College in Bjelovar

1.11. Further reading (at the time of proposing the study programme)

- Robert Manger: *Strukture podataka i algoritmi*, Element, Zagreb, 2014

1.12. Number of copies of required reading compared to the number of students currently taking the course

Title	*Number of copies*	*Number of students*
Kusalić, Domagoj: *Napredno programiranje i algoritmi u C-u i C++-u*, 5th edition, Element, Zagreb, 2014	6	30
Dario Vidić: Presentations for lectures and practical sessions in Data Structures and Algorithms, Technical College in Bjelovar	0, available online	30

1.13. *Methods of quality monitoring aimed at ensuring the acquisition of output knowledge, skills and competences*

Conducting student surveys and evaluating the data in accordance with the results of the Student Survey Committee and the analyses of exam results in accordance with the data from the ISVU data warehouse.

1.14. Possibility of course instruction in a foreign language

English

GENERAL INFORMATION							
Course title	**Database Systems**						
Programme of study	Undergraduate professional programme of study in Computer Science						
Course status	Compulsory						
Year	2nd						
Semester	3rd						
Calculation of ECTS credits	Lectures	Practical sessions		Preparation for practical sessions	Seminar paper	Self-study	Total
		APS	LPS				
	30	0	30	30	0	90	180
ECTS credits and forms of instruction	ECTS student workload coefficient				6		
	Contact hours (L+PS+S)				L	PS	S
					30	30	

1. COURSE DESCRIPTION

1.1. Course objectives

1. To understand database management systems.
2. To understand the design of relational databases, the entity-relationship model and relational algebra.
3. To learn the SQL query language and the fundamentals of database protection.

1.2. Prerequisites

None.

1.3. Expected learning outcomes

Upon completion of the course students will be able to:

1. design and create a database on their own,
2. implement simple and complex queries using the SQL,
3. classify types of database management systems,
4. apply relational algebra as query language,
5. apply SQL in the creation, definition and modification of a database,
6. explain problems related to database security.

1.4. Course content

1. **Introduction to databases**
   Defining the fundamental concepts. Database architecture. Database management systems.
2. **Relational data model**
   Relational schema. Operations in the relational data model. Introduction to SQL. Relational algebra. Fundamentals of SQL language use.
   Introduction. Fundamental objects in SQl language. Statement format. Types of data. Data and relation definition statements. Basic SQL statements for data management. Statements for transfer of data to and from an operating system file.
3. **The *SELECT* statement**
   Simple queries. Expression. Defining conditions. Aggregate functions. Joined relations. SQL statements containing conditions with subqueries. Grouping results. Specifying conditions for a group

of records. Order of results. Storing results in a temporary relation. Determining union of relations using the *select* statement.

4. ***INSERT*, *DELETE*, *UPDATE* statements**
   Ways of using the *INSERT* statement. *INSERT* statement. *DELETE* statement. *UPDATE* statement.
5. ***NULL* values**

1.5. Forms of instruction	☒ lectures ☐ seminars and workshops ☒ practical sessions ☐ distance learning ☐ field work	☐ individual work ☐ multimedia and web ☒ laboratory sessions ☐ mentorship ☐ other _____
1.6. Comments		

1.7. Students' obligations

2 preliminary exams
2 homework assignments
15 laboratory sessions

1.8. Students' performance monitoring

Class attendance	**0.5**	Class participation		Seminar paper		Experimental work	0
Written exam	**2.5**	Oral exam	**1**	Essay		Research	
Project		Continuous assessment	**1**	Class report		Practical work	**1**
Portfolio							

1.9. Grading and evaluation of student work in class and in final exam

Homework assignments: 5 x 2 points = 10 points
Work in laboratory sessions: 30 points
Two preliminary exams/exam: 60 points (a minimum of 20 points is required),
1st preliminary exam: 30 points (a minimum of 10 points is required),
2nd preliminary exam: 30 points (a minimum of 10 points is required).

A maximum of 100 points can be scored in the course. Students pass the written exam if they have achieved a total of 50 points doing their homework assignments, laboratory practice and preliminary exams. Students who have passed the written exam by passing the preliminary exams are exempt from taking the oral exam, except if they wish to improve their grade.

Grading and evaluation of student work after the course:
Written exam: 100 points
Oral exam

Students pass the written exam if they have scored 50 points. After they have passed the written exam, students take the oral exam.

WRITTEN EXAM GRADING:
[50, 63 > points - sufficient (2)
[63, 75 > points - good (3)
[75, 87 > points - very good (4)
[87, 100] points - excellent (5)

1.10. Required reading (at the time of proposing the study programme)

- Željko Knok: Presentations for lectures and practical sessions in the Database Systems course, Technical College in Bjelovar

1.11. Further reading (at the time of proposing the study programme)

- C. J. Date: *An Introduction to Database Systems*, 8th edition, Addison Wesley, Boston, 2006
- J. D. Ullman, J. Widom: *A First Course in Database Systems*, Prentice-Hall, 2008

1.12. Number of copies of required reading compared to the number of students currently taking the course

Title	*Number of copies*	*Number of students*
Željko Knok: Presentations for lectures and practical sessions in the Database Systems course, Technical College in Bjelovar	0, available online	30

1.13. Methods of quality monitoring aimed at ensuring the acquisition of output knowledge, skills and competences

Conducting student surveys and evaluating the data in accordance with the results of the Student Survey Committee and the analyses of exam results in accordance with the data from the ISVU data warehouse.

1.14. Possibility of course instruction in a foreign language

English

GENERAL INFORMATION							
Course title	**Technical English 3**						
Programme of study	Undergraduate professional programme of study in Computer Science						
Course status	Compulsory						
Year	2nd						
Semester	3rd						
Calculation of ECTS credits	Lectures	Practical sessions		Preparation for practical sessions	Seminar paper	Self-study	Total
		APS	LPS				
	15	30				15	60
ECTS credits and forms of instruction	ECTS student workload coefficient				2		
	Contact hours (L+PS+S)				L	PS	S
					15	30	

1. COURSE DESCRIPTION
1.1. Course objectives
1. To develop students' ability to use the English language related to specific technical fields.
1.2. Prerequisites
Previous enrolment in **Technical English 2**.
1.3. Expected learning outcomes
Upon completion of the course students will be able to use the English language to: 1. describe procedures and requirements related to occupational health and safety, 2. describe the differences between regulations and standards and give examples, 3. describe automated systems, measurable parameters, readings and approximate values, 4. explain testing procedures, conduction of experiments and describe the predicted outcomes of testing, 5. demonstrate mastery of more complex grammatical structures.
1.4. Course content
1. Occupational health and safety Description of occupational health and safety measures. Protective equipment. Regulations and standards. Written instructions. Protective measures in office work and work with computers. Grammar focus: differences between colloquial and written style, imperative, conditional sentences. **2. Automatic control** Describing automated systems. Describing measurable parameters. Giving approximate values. Describing graphs. The role of computers in automation. Grammar focus: passive structures. **3. Tests and experiments** Describing tests and experiments. Comparison of results and expectations. Assumptions. Tests and experiments in programming. Grammar focus: types of nouns, plural of nouns, compounds, articles.

1.5. Forms of instruction	☒ lectures ☐ seminars and workshops ☒ practical sessions	☐ individual work ☐ multimedia and web ☐ laboratory sessions

	☐ distance learning ☐ field work	☐ mentorship ☐ other _____
1.6. Comments		
1.7. Students' obligations		
2 preliminary exams or the written exam oral exam		
1.8. Students' performance monitoring		

Class attendance		Class participation	**0.5**	Seminar paper		Experimental work	
Written exam	**1**	Oral exam	**0.5**	Essay		Research	
Project		Continuous assessment		Class report		Practical work	
Portfolio							

1.9. Grading and evaluation of student work in class and in final exam

Written exam/two preliminary exams: 50 points (25 points per each preliminary exam)
Oral exam: 25 points
Class work (practice): 25 points

To pass the written exam, a score of 25 points is required. Passing the written exam (by taking the preliminary exams or the final examination) is a prerequisite for taking the oral exam.
The final grade is based on scores attained in the written exam, oral exam and class work as follows:

50 points - 60 points = sufficient (2)
61 points - 74 points = good (3)
75 points - 87 points = very good (4)
88 points - 100 points = excellent (5)

1.10. Required reading (at the time of proposing the study programme)

1. Ibbotson, Mark: *Cambridge English for Engineering*, Cambridge University Press, Cambridge, 2008

1.11. Further reading (at the time of proposing the study programme)

1. Murphy, Raymond: *English Grammar in Use*, Cambridge University Press, Cambridge, 2004

1.12. Number of copies of required reading compared to the number of students currently taking the course

Title	*Number of copies*	*Number of students*
Ibbotson, Mark: *Cambridge English for Engineering*, Cambridge University Press, Cambridge, 2008	6	30

1.13. Methods of quality monitoring aimed at ensuring the acquisition of output knowledge, skills and competences
Conducting student surveys and evaluating the data in accordance with the results of the Student Survey Committee and the analyses of exam results in accordance with the data from the ISVU data warehouse.
a. Possibility of course instruction in a foreign language
English

GENERAL INFORMATION							
Course title	**Computer Architectures**						
Programme of study	Undergraduate professional programme of study in Computer Science						
Course status	Compulsory						
Year	2nd						
Semester	4th						
Calculation of ECTS credits	Lectures	Practical sessions		Preparation for practical sessions	Seminar paper	Self-study	Total
		APS	LPS				
	15		30	30		45	120
ECTS credits and forms of instruction	ECTS student workload coefficient				4		
	Contact hours (L+PS+S)				L	PS	S
					15	30	

1. COURSE DESCRIPTION

1.1. Course objectives

1. Familiarising students with computer architecture.
2. Familiarising students with the architecture of Atmel's ATMega16 microcomputer.
3. Learning to programme in the AtmelStudio 6.0 environment using the C/C++ programming language.
4. Learning to connect a microcomputer to a personal computer and smartphone by serial communication.

1.2. Prerequisites

None.

1.3. Expected learning outcomes

Upon completion of the course students will be able to:

1. recognise differences between existing computer architectures,
2. assess when it is possible and cost effective to use microcomputers,
3. connect a sensor and actuator to a microcomputer and send data to them,
4. connect various modules to a microcomputer,
5. design a microcomputer system (electronics + microcomputer programming in Atmel Studio 6.0),
6. connect a microcomputer system to a PC.

1.4. Course content

1. Fundamental concepts and computer architecture
Historical development of computers. Differences between microcomputers, microcontrollers and microprocessors.
CPU. Computer architectures (CISC, RISC). Microcomputers' instruction execution. Von Neumann's model of computer architecture. Microcomputer architecture.
2. The Atmel ATMega16 Microcontroller
Producers of microcontrollers. The RISC architecture of the Atmel ATMega16 microcontroller. Status register. Instruction execution. RAM and EEPROM memory. Input/output registers. Clock rate. *Reset*.

Watchdog. Interrupts. Digital inputs and outputs. Counters and timers. PWM. Analog-to-digital converters and analog inputs. USART (serial communication). I2C communication. SPI communication. External interrupts.

3. Programming a microcontroller

Programming a microcontroller in the assembler and C/C++ programming languages. Instruction set of the ATMega16 microcontroller. Machine code. Interrupt routines. Functions. The Atmel Studio 6.0 programming environment. Fuse bits. Lock bits. In-System Programming.

4. Connecting external circuits and electronic devices to a microcontroller

Push buttons. LED diodes. LCD display. Potentiometer. NTC resistor. Numeric display. MAX232. FT232. Optocoupler. Transistor as a switch. Relay. Buzzer. Temperature sensors: DS18B20 and LM35. Bluetooth module. Graphic display. GSM module. Servo motor. Ultrasound sensor.

5. Microcomputer system management using a computer or smartphone application

1.5. Forms of instruction	☒ lectures ☐ seminars and workshops ☒ practical sessions ☐ distance learning ☐ field work	☒ individual work ☐ multimedia and web ☒ laboratory sessions ☐ mentorship ☐ other _____
1.6. Comments		

1.7. Students' obligations

10 laboratory sessions
project work

1.8. Students' performance monitoring

Class attendance	**0.25**	Class participation		Seminar paper		Experimental work	**1**
Written exam	**0.5**	Oral exam	**0.5**	Essay		Research	
Project		Continuous assessment		Class report		Practical work	**1.75**
Portfolio							

1.9. Grading and evaluation of student work in class and in final exam

Laboratory sessions: 10 x 3 points = 30 points
Project work: 70 points

Project teams of up to 4 members will be formed during lectures and laboratory sessions. The teams will have their leaders. Each project team will get to choose an assignment which includes the design and programming of a microcomputer system. The project team will be provided with support from the course instructor and laboratory assistant. Upon completion of the assignment, the project team is expected to prepare complete technical documentation related to the microcomputer system, which is to be drafted in accordance with the template published on the course website.

Upon completion of the assignment the project team is required to defend their work before the course instructor. After the defence, the course instructor awards grades to project team members. By completing the technical documentation and design of a microcomputer system students can earn a maximum of 80 points. After applying for the exam, students take the oral exam.

WRITTEN EXAM GRADING:
[50, 63 > points - sufficient (2)

[63, 75 > points - good (3)
[75, 87 > points - very good (4)
[87, 100] points - excellent (5)

1.10. Required reading (at the time of proposing the study programme)

1. Vrhovski, Zoran; Miletić, Marko: Mikroračunala - Programiranje mikrokontrolera porodice Atmel u programskom okruženju Atmel Studio 6, Technical College in Bjelovar, Bjelovar, 2014
2. Vrhovski, Zoran: Presentations for lectures in Computer Architectures, Technical College in Bjelovar
3. ATMEL: 8-bit AVR Microcontroller with 16K Bytes In-System Programmable Flash – ATMEGA16, www.atmel.com/Images/doc2466.pdf (available: 20 February 2016)

1.11. Further reading (at the time of proposing the study programme)

1. F. Barrett, Steven; Pack, Daniel; Thornton, Mitchell: *Atmel AVR Microcontroller Primer: Programming and Interfacing*, Morgan &Claypool Publishers, Thornton, 2007

1.12. Number of copies of required reading compared to the number of students currently taking the course

Title	*Number of copies*	*Number of students*
Vrhovski, Zoran; Miletić, Marko: Mikroračunala - Programiranje mikrokontrolera porodice Atmel u programskom okruženju Atmel Studio 6, Technical College in Bjelovar, Bjelovar, 2014	6	30
Vrhovski, Zoran: Presentations for lectures in Computer Architectures, Technical College in Bjelovar	0, available online	30
ATMEL: 8-bit AVR Microcontroller with 16K Bytes In-System Programmable Flash – ATMEGA16	0, available online	30

1.13. Methods of quality monitoring aimed at ensuring the acquisition of output knowledge, skills and competences

Conducting student surveys and evaluating the data in accordance with the results of the Student Survey Committee and the analyses of exam results in accordance with the data from the ISVU data warehouse.

1.14. Possibility of course instruction in a foreign language

English

GENERAL INFORMATION							
Course title	**Software Engineering**						
Programme of study	Undergraduate professional programme of study in Computer Science						
Course status	Compulsory						
Year	2nd						
Semester	4th						
Calculation of ECTS credits	Lectures	Practical sessions		Preparation for practical sessions	Seminar paper	Self-study	Total
		APS	LPS				
	30	0	15	10	0	65	120
ECTS credits and forms of instruction	ECTS student workload coefficient				4		
	Contact hours (L+PS+S)				L	PS	S
					30	15	

1. COURSE DESCRIPTION

1.1. Course objectives

1. Understanding steps in software development.
2. Acquisition of basic knowledge on a systematic, disciplined and measurable approach to software development, implementation and maintenance.

1.2. Prerequisites

None.

1.3. Expected learning outcomes

Upon completion of the course students will be able to:

1. explain the reasons for applying engineering principles in software development,
2. describe software process models,
3. acquire skill at working with at least one software modelling tools,
4. test simple programs.

1.4. Course content

1. **Introduction to software engineering**
   Software life cycle. Software life cycle models.
2. **Software processes and other models**
   Software process model. Data flow diagram. Petri net. Object model. Sequence diagram. Hierarchy diagram. Control flow graph. State diagram. UML diagrams.
3. **Software project management**
   Approaches to management. Team approach. Critical practice. Maturity model. Personal software process. Earned value analysis. Defect tracking. Post-mortem analysis.
4. **Software project planning**
   Project planning. Assigning responsibilities. The program evaluation and review technique. Software cost estimation.
5. **Software measurement**
   Measurement. Software measurement theory. Software metrics. Process metrics.

6. **Risk management and analysis**
   Risk identification. Risk assessment. Risk exposure. Risk reduction. Risk management plan.
7. **Software quality assurance**
   Formal inspection and technical inspection. Software reliability. IEEE standards for software quality assurance.
8. **Requirements**
   Object model. Data flow modelling. Behaviour modelling. Dictionary. System diagram. IEEE requirements specification standard.
9. **Software design**
   Stages in the design process. Design concepts. Measuring cohesion. Measuring coupling. Requirements traceability.
10. **Software testing**
    Fundamentals of software testing. Test coverage criteria. Dana flow testing. Random testing. Boundary testing.
11. **Object-oriented development, metrics and testing**
    Inheritance and polymorphism. Object identification. Association identification. Identification of multiplicity. Object-oriented design metrics. MOOD metrics. MM testing. Function pair coverage.

1.5. Forms of instruction	☒ lectures ☐ seminars and workshops ☒ practical sessions ☐ distance learning ☐ field work	☐ individual work ☐ multimedia and web ☒ laboratory sessions ☐ mentorship ☐ other _____
1.6. Comments		

1.7. Students' obligations

2 preliminary exams
2 homework assignments
6 laboratory sessions

1.8. Students' performance monitoring

Class attendance	**0.5**	Class participation		Seminar paper		Experimental work	0
Written exam	**2**	Oral exam	**1**	Essay		Research	
Project		Continuous assessment	**0.5**	Class report		Practical work	0
Portfolio							

1.9. Grading and evaluation of student work in class and in final exam

Homework assignments: 5 x 2 points = 10 points
Work in laboratory sessions: 10 points
Two preliminary exams/exam: 80 points (a minimum of 30 points is required),
1st preliminary exam: 40 points (a minimum of 15 points is required),
2nd preliminary exam: 40 points (a minimum of 15 points is required).

A maximum of 100 points can be scored in the course. Students pass the written exam if they have achieved a total of 50 points doing their homework assignments, laboratory practice and preliminary exams. Students who have passed the written exam by passing the preliminary exams are exempt from taking the oral exam, except if they wish to improve their grade.

Grading and evaluation of student work after the course:
Written exam: 100 points
Oral exam

Students pass the written exam if they have scored 50 points. After they have passed the written exam, students take the oral exam.

WRITTEN EXAM GRADING:
[50, 63 > points - sufficient (2)
[63, 75 > points - good (3)
[75, 87 > points - very good (4)
[87, 100] points - excellent (5)

1.10. Required reading (at the time of proposing the study programme)

- Mihael Kukec: Presentations for lectures and practical sessions in Software Engineering, Technical College in Bjelovar

1.11. Further reading (at the time of proposing the study programme)

- David A. Gustafson: *Schaum's Outlines of Software Engineering*, McGraw-Hill, 2002
- Roger S. Pressman: *Software Engineering: A Practitioner's Approach*, McGraw-Hill, 2014
- Ian Sommerville: *Software Engineering*, Addison-Wesley, 2011

1.12. Number of copies of required reading compared to the number of students currently taking the course

Title	*Number of copies*	*Number of students*
Mihael Kukec: Presentations for lectures and practical sessions in Software Engineering, Technical College in Bjelovar	0, available online	30

1.13. Methods of quality monitoring aimed at ensuring the acquisition of output knowledge, skills and competences

Conducting student surveys and evaluating the data in accordance with the results of the Student Survey Committee and the analyses of exam results in accordance with the data from the ISVU data warehouse.

1.14. Possibility of course instruction in a foreign language

English

GENERAL INFORMATION							
Course title	**Operating Systems**						
Programme of study	Undergraduate professional programme of study in Computer Science						
Course status	Compulsory						
Year	2nd						
Semester	4th						
Calculation of ECTS credits	Lectures	Practical sessions		Preparation for practical sessions	Seminar paper	Self-study	Total
		APS	LPS				
	30		15	0	0	105	150
ECTS credits and forms of instruction	ECTS student workload coefficient				5		
	Contact hours (L+PS+S)				L	PS	S
					30	15	

1. COURSE DESCRIPTION

1.1. Course objectives

1. Mastering the fundamentals of operating systems.
2. Understanding the purpose of operating systems.

1.2. Prerequisites

None.

1.3. Expected learning outcomes

Upon completion of the course students will be able to:

1. explain the operation and structure of operating systems,
2. explain the concept of addresses,
3. explain the concept of threads,
4. explain the principles of memory and disc management.

1.4. Course content

1. **Introduction**
   History of operating systems. Tasks of operating systems.
2. **Simple computer model**
   The Von Neumann computer model. Description of computer components. Thread.
3. **Performing input/output operations, interrupts**
   Connecting devices to a computer. Busy waiting and processor interrupts. Direct memory access. Multiprocessor systems.
4. **Mutual exclusion in multithreaded systems**
   Fundamental concepts. Multithreaded task completion. Multithreading model. Mutual exclusion. Hardware-supported mutual exclusion.
5. **Kernel of an operating system**
   Kernel data structure. Kernel functions. Realisation of the kernel in a multiprocessor system.
6. **Inter-thread communication and the concept of monitor**
   Producer and consumer problem. Deadlock. The concept of monitor.
7. **Analysis of time features of an operating system**

Dynamic behaviour of an operating system. Basic ways of assigning processors to threads.

8. **Memory management**
   Static and dynamic memory management. Paging.
9. **File subsystem**
   Discs. File systems. The role of data buffer.
10. **Inter-process communication**
11. **Inter-thread communication. Communication in a distributed system.**

1.5. Forms of instruction	☒ lectures ☐ seminars and workshops ☐ distance learning ☐ field work	☐ individual work ☐ multimedia and web ☒ laboratory sessions ☐ mentorship ☐ other _____
1.6. Comments		

1.7. Students' obligations

2 preliminary exams
2 homework assignments
4 laboratory sessions

1.8. Students' performance monitoring

Class attendance	**0.5**	Class participation		Seminar paper		Experimental work	0
Written exam	**2.5**	Oral exam	**1**	Essay		Research	
Project		Continuous assessment	**0.5**	Class report		Practical work	**0.5**
Portfolio							

1.9. Grading and evaluation of student work in class and in final exam

Homework assignments: 5 x 2 points = 10 points
Work in laboratory sessions: 10 points
Two preliminary exams/exam: 80 points (a minimum of 30 points is required),
1st preliminary exam: 40 points (a minimum of 15 points is required),
2nd preliminary exam: 40 points (a minimum of 15 points is required).

A maximum of 100 points can be scored in the course. Students pass the written exam if they have achieved a total of 50 points doing their homework assignments, laboratory practice and preliminary exams. Students who have passed the written exam by passing the preliminary exams are exempt from taking the oral exam, except if they wish to improve their grade.

Grading and evaluation of student work after the course:
Written exam: 100 points
Oral exam

Students pass the written exam if they have scored 50 points. After they have passed the written exam, students take the oral exam.

WRITTEN EXAM GRADING:
[50, 63 > points - sufficient (2)
[63, 75 > points - good (3)
[75, 87 > points - very good (4)
[87, 100] points - excellent (5)

1.10. Required reading (at the time of proposing the study programme)

- Zoran Vrhovski: Presentations for lectures and practical sessions in Operating Systems, Technical College in Bjelovar
- Leo Budin, Marin Golub, Domagoj Jakobović, Leonardo Jelenković: *Operacijski sustavi*, Element, Zagreb, 2010

1.11. Further reading (at the time of proposing the study programme)

- A. Silberschatz, P. B. Galvin, G. Gagne: *Operating System Concepts*, John Wiley&Sons, 2003

1.12. Number of copies of required reading compared to the number of students currently taking the course

Title	*Number of copies*	*Number of students*
Zoran Vrhovski: Presentations for lectures and practical sessions in Operating Systems, Technical College in Bjelovar	0, available online	30
Leo Budin, Marin Golub, Domagoj Jakobović, Leonardo Jelenković: *Operacijski sustavi*, Element, Zagreb, 2010	6	30

1.13. Methods of quality monitoring aimed at ensuring the acquisition of output knowledge, skills and competences

Conducting student surveys and evaluating the data in accordance with the results of the Student Survey Committee and the analyses of exam results in accordance with the data from the ISVU data warehouse.

1.14. Possibility of course instruction in a foreign language

English

GENERAL INFORMATION							
Course title	**Programming in JAVA**						
Programme of study	Undergraduate professional programme of study in Computer Science						
Course status	Compulsory						
Year	2nd						
Semester	4th						
Calculation of ECTS credits	Lectures	Practical sessions		Preparation for practical sessions	Seminar paper	Self-study	Total
		APS	LPS				
	30	0	30	15	0	105	180
ECTS credits and forms of instruction	ECTS student workload coefficient				6		
	Contact hours (L+PS+S)				L	PS	S
					30	30	

1. COURSE DESCRIPTION

1.1. Course objectives

1. Acquisition of basic knowledge of programming in the JAVA programming language.
2. Mastering object-oriented programming in developing Java applications.
3. Teaching students to programme simple programs.
4. Correlating knowledge of other programming languages with Java.

1.2. Prerequisites

None.

1.3. Expected learning outcomes

Upon completion of the course students will be able to:

1. explain the structure and model of the Java programming language,
2. develop simple software in the Java programming language,
3. apply object-oriented programming in the development of Java applications,
4. recognise the advantages and shortcomings of the Java programming language.

1.4. Course content

1. **Introduction**
   Historical development. Development of Java programming.
2. **Introduction to development tools**
   The *Eclipse* integrated development environment. Creating the *HelloWorld* project.
3. **Fundamental structures of the Java programming language**
   Variables and expressions. Loops and flow control. The *If* condition. *While*, *Do While*, *For* loops.
4. **Arrays**
   One-dimensional arrays. Multidimensional arrays. Operations on arrays.
5. **Sequences of characters**
   Methods for working with sequences of characters.
6. **Classes**
   Class declaration. Object instantiation. Constructors. Destructors. Object destruction. Static members.
   Inheritance. Polymorphism. Abstract classes.

7. Exceptions
Exceptions in Java. Classification of exceptions. Processing multiple exceptions.

8. Input and output streams
Binary streams. Character streams.

1.5. Forms of instruction	☒ lectures ☐ seminars and workshops ☒ practical sessions ☐ distance learning ☐ field work	☐ individual work ☐ multimedia and web ☒ laboratory sessiions ☐ mentorship ☐ other _____
1.6. Comments		

1.7. Students' obligations

2 preliminary exams
2 homework assignments
15 laboratory sessions

1.8. Students' performance monitoring

Class attendance	**0.5**	Class participation		Seminar paper		Experimental work	0
Written exam	**2.5**	Oral exam	**1.5**	Essay		Research	
Project		Continuous assessment	**0.5**	Class report		Practical work	**1**
Portfolio							

1.9. Grading and evaluation of student work in class and in final exam

Homework assignments: 5 x 2 points = 10 points
Work in laboratory sessions: 30 points
Two preliminary exams/exam: 60 points (a minimum of 20 points is required),
1st preliminary exam: 30 points (a minimum of 10 points is required),
2nd preliminary exam: 30 points (a minimum of 10 points is required).

A maximum of 100 points can be scored in the course. Students pass the written exam if they have achieved a total of 50 points doing their homework assignments, laboratory practice and preliminary exams. Students who have passed the written exam by passing the preliminary exams are exempt from taking the oral exam, except if they wish to improve their grade.

Grading and evaluation of student work after the course:
Written exam: 100 points
Oral exam

Students pass the written exam if they have scored 50 points. After they have passed the written exam, students take the oral exam.

WRITTEN EXAM GRADING:
[50, 63 > points - sufficient (2)
[63, 75 > points - good (3)
[75, 87 > points - very good (4)
[87, 100] points - excellent (5)

1.10. Required reading (at the time of proposing the study programme)

- Igor Petrović: Presentations for lectures and practical sessions in Programming in JAVA, Technical College in Bjelovar
- Yakov Fain: *Programiranje Java*, Kompjuter biblioteka, Zagreb, 2015

1.11. Further reading (at the time of proposing the study programme)

1.12. Number of copies of required reading compared to the number of students currently taking the course

Title	*Number of copies*	*Number of students*
Igor Petrović: Presentations for lectures and practical sessions in Programming in JAVA, Technical College in Bjelovar	0, available online	30
Yakov Fain: *Programiranje Java*, Kompjuter biblioteka, Zagreb, 2015	6	30

1.13. Methods of quality monitoring aimed at ensuring the acquisition of output knowledge, skills and competences

Conducting student surveys and evaluating the data in accordance with the results of the Student Survey Committee and the analyses of exam results in accordance with the data from the ISVU data warehouse.

1.14. Possibility of course instruction in a foreign language

English

GENERAL INFORMATION							
Course title	**C# Programming 1**						
Programme of study	Undergraduate professional programme of study in Computer Science						
Course status	Compulsory						
Year	2nd						
Semester	4th						
Calculation of ECTS credits	Lectures	Practical sessions		Preparation for practical sessions	Seminar paper	Self-study	Total
		APS	LPS				
	30	0	30	30	0	60	150
ECTS credits and forms of instruction	ECTS student workload coefficient				5		
	Contact hours (L+PS+S)				L	PS	S
					30	30	

1. COURSE DESCRIPTION

1.1. Course objectives

1. Acquisition of basic knowledge of programming in the c# programming language.
2. Learn the object-oriented paradigms of the c# programming language.
3. Familiarise students with the .NET environment.

1.2. Prerequisites

None.

1.3. Expected learning outcomes

1. Being able to describe the structure and model of the C# programming language.
2. Designing simple software in the C# programming language.
3. Designing a simple *Windows Form* application.
4. Being able to describe the object-oriented approach to programming.

1.4. Course content

1. **Introduction**
   Introduction to C#. Introduction to .NET. The *Visual Studio* runtime environment.
2. **Fundamental programming techniques**
   Opening a new project. Program structure. Commenting and code readability. Variables. Operators. Loops and flow control.
3. **Abstraction using classes**
   Defining class. Value types and reference types. Constructors. Defining methods. Encapsulation.
4. **Sequences and lists**
   Sequences. List<T>. Collections and polymorphism. Structures.
5. **Character sequences**
   String and *Char* types. Formatting data for display. Differentiating between culture settings. Working with a text. Character encoding.
6. ***Windows Forms***
   Designing a user interface. Working with controls. Events. Event processing.
7. **Files and data streams**
   Working with directories and files. Text files. Data streams.

1.5. Forms of instruction	☒ lectures ☐ seminars and workshops ☒ practical sessions ☐ distance learning ☐ field work	☐ individual work ☐ multimedia and web ☒ laboratory sessions ☐ mentorship ☐ other _____
1.6. Comments		

1.7. Students' obligations

2 preliminary exams
2 homework assignments
15 laboratory sessions

1.8. Students' performance monitoring

Class attendance	0	Class participation		Seminar paper		Experimental work	0
Written exam	0	Oral exam	0	Essay		Research	
Project		Continuous assessment	0	Class report		Practical work	0
Portfolio							

1.9. Grading and evaluation of student work in class and in final exam

Homework assignments: 5 x 2 points = 10 points
Work in laboratory sessions: 30 points
Two preliminary exams/exam: 60 points (a minimum of 20 points is required),
1st preliminary exam: 30 points (a minimum of 10 points is required),
2nd preliminary exam: 30 points (a minimum of 10 points is required).

A maximum of 100 points can be scored in the course. Students pass the written exam if they have achieved a total of 50 points doing their homework assignments, laboratory practice and preliminary exams. Students who have passed the written exam by passing the preliminary exams are exempt from taking the oral exam, except if they wish to improve their grade.

Grading and evaluation of student work after the course:
Written exam: 100 points
Oral exam

Students pass the written exam if they have scored 50 points. After they have passed the written exam, students take the oral exam.

WRITTEN EXAM GRADING:
[50, 63 > points - sufficient (2)
[63, 75 > points - good (3)
[75, 87 > points - very good (4)
[87, 100] points - excellent (5)

1.10. Required reading (at the time of proposing the study programme)

- Bruno Trstenjak: Presentations for lectures and practical sessions in the C# Programming 1 course, Technical College in Bjelovar
- Ian Griffiths, Matthew Adams, Jesse Liberty: *Programiranje C# 4.0*, O'Reilly Media, 2011

1.11. Further reading (at the time of proposing the study programme)

1.12. Number of copies of required reading compared to the number of students currently taking the course

Title	*Number of copies*	*Number of students*
Bruno Trstenjak: Presentations for lectures and practical sessions in the C# Programming 1 course, Technical College in Bjelovar	0, available online	30
Ian Griffiths, Matthew Adams, Jesse Liberty: *Programiranje C# 4.0*, O'Reilly Media, 2011	6	30

1.13. Methods of quality monitoring aimed at ensuring the acquisition of output knowledge, skills and competences

Conducting student surveys and evaluating the data in accordance with the results of the Student Survey Committee and the analyses of exam results in accordance with the data from the ISVU data warehouse.

1.14. Possibility of course instruction in a foreign language

English

GENERAL INFORMATION							
Course title	**Computer Security**						
Programme of study	Undergraduate professional programme of study in Computer Science						
Course status	Compulsory						
Year	2nd						
Semester	4th						
Calculation of ECTS credits	Lectures	Practical sessions		Preparation for practical sessions	Seminar paper	Self-study	Total
		APS	LPS				
	30	0	15	15	0	60	120
ECTS credits and forms of instruction	ECTS student workload coefficient				4		
	Contact hours (L+PS+S)				L	PS	S
					30	15	

1. COURSE DESCRIPTION

1.1. Course objectives

1. Familiarising students with the concept of information system protection, methods of threat prevention and risk identification/detection.
2. Getting students to understand the risks involved in networked systems.
3. Getting students to understand the weaknesses of operating systems.
4. Teach students about approaches to operation system security enhancement.

1.2. Prerequisites

None.

1.3. Expected learning outcomes

Upon completion of the course students will be able to:

1. explain the shortcomings of operating systems, software, databases, networks and mobile devices,
2. enhance security of a specific operating system,
3. apply security enhancement when designing their own applications with a view of ensuring more secure data exchange.

1.4. Course content

1. **Introduction to computer security**
   Fundamental concepts, security threats, security goals.
2. **Operating system security**
   Access control, authorisation, user authentication, memory protection. Biometric identification.
3. **Software security**
   Malicious code analysis and defence (viruses, spyware, rootkits). Web security: XSS and XSRF attacks and defences.
4. **Database security**
   Access control, privacy.
5. **Network security**
   IPSec, SSL/TSL, DDoS attacks, DNS, network firewalls.
6. **Mobile device security**
7. **Fundamentals of cryptography**
   Encryption, authentication, random numbers.

1.5. Forms of instruction	☒ lectures ☐ seminars and workshops ☒ practical sessions ☐ distance learning ☐ field work	☐ individual work ☐ multimedia and web ☒ laboratory sessions ☐ mentorship ☐ other _____
1.6. Comments		

1.7. Students' obligations

2 preliminary exams
2 homework assignments
6 laboratory sessions

1.8. Students' performance monitoring

Class attendance	**0.5**	Class participation		Seminar paper		Experimental work	0
Written exam	**2**	Oral exam	**1**	Essay		Research	
Project		Continuous assessment	**0.5**	Class report		Practical work	0
Portfolio							

1.9. Grading and evaluation of student work in class and in final exam

Homework assignments: 5 x 2 points = 10 points
Work in laboratory sessions: 10 points
Two preliminary exams/exam: 80 points (a minimum of 30 points is required),
1st preliminary exam: 40 points (a minimum of 15 points is required),
2nd preliminary exam: 40 points (a minimum of 15 points is required).

A maximum of 100 points can be scored in the course. Students pass the written exam if they have achieved a total of 50 points doing their homework assignments, laboratory practice and preliminary exams. Students who have passed the written exam by passing the preliminary exams are exempt from taking the oral exam, except if they wish to improve their grade.

Grading and evaluation of student work after the course:
Written exam: 100 points
Oral exam

Students pass the written exam if they have scored 50 points. After they have passed the written exam, students take the oral exam.

WRITTEN EXAM GRADING:
[50, 63 > points - sufficient (2)
[63, 75 > points - good (3)
[75, 87 > points - very good (4)
[87, 100] points - excellent (5)

1.10. Required reading (at the time of proposing the study programme)

- Robert Herčeki: Presentations for lectures and practical session in Computer Security, Technical College in Bjelovar

1.11. Further reading (at the time of proposing the study programme)

- Dieter Gollmann: *Computer Security*, 2nd edition, Wiley, 2005
- Miroslav Bača: *Uvod u računalnu sigurnost*, Narodne novine, Zagreb, 2004

1.12. Number of copies of required reading compared to the number of students currently taking the course

Title	*Number of copies*	*Number of students*
Robert Herčeki: Presentations for lectures and practical session in Computer Security, Technical College in Bjelovar	0, available online	30

1.13. Methods of quality monitoring aimed at ensuring the acquisition of output knowledge, skills and competences

Conducting student surveys and evaluating the data in accordance with the results of the Student Survey Committee and the analyses of exam results in accordance with the data from the ISVU data warehouse.

1.14. Possibility of course instruction in a foreign language

English

GENERAL INFORMATION							
Course title	**Technical English 4**						
Programme of study	Undergraduate professional programme of study in Computer Science						
Course status	Compulsory						
Year	2nd						
Semester	4th						
Calculation of ECTS credits	Lectures	Practical sessions		Preparation for practical sessions	Seminar paper	Self-study	Total
		APS	LPS				
	15	30				15	60
ECTS credits and forms of instruction	ECTS student workload coefficient				2		
	Contact hours (L+PS+S)				L	PS	S
					15	30	

1. COURSE DESCRIPTION
1.1. Course objectives
To render students capable of using the English language related to the technical field.
1.2. Prerequisites
Previous enrolment in **Technical English 3**.
1.3. Expected learning outcomes
Upon completion of the course students will be able to: 1. describe physical forces in the English language, 2. use English to describe differences between fossil fuels and renewable energy sources, 3. communicate in English via e-mail, 4. use English to draft a curriculum vitae and job application in standard international formats, 5. prepare and give a 10-minute presentation in English on one of the given topics, 6. use more complex grammatical structures.
1.4. Course content
1. Energy Energy. Fossil fuels and renewable energy sources. Describing forces. Describing possibility and restrictions. Grammar focus: reported speech, gerund vs. infinitive. **2. Writing formal and informal e-mails** Levels of formality. Enquiries. Exchanging information. Deadlines. Confirming deadlines and arrangements. **3. Curriculum vitae and job application** Writing a curriculum vitae and job application in the English language. **4. Presentation skills in the English language** Preparing a short, structured presentation in English on a given topic related to the technical field.

1.5. Forms of instruction		
	☒ lectures	☐ individual work
	☐ seminars and workshops	☐ multimedia and web
	☒ practical sessions	☐ laboratory sessions
	☐ distance learning	☐ mentorship

	☐ field work	☐ other _____
1.6. Comments		
1.7. Students' obligations		
2 preliminary exams or the written exam presentation		

1.8. Students' performance monitoring

Class attendance		Class participation	**0.5**	Seminar paper		Experimental work	
Written exam	**1**	Oral exam		Essay		Research	
Project		Continuous assessment		Class report		Practical work	
Portfolio		Presentation	**0.5**				

1.9. Grading and evaluation of student work in class and in final exam

Written exam/two preliminary exams: 50 points (25 points per each preliminary exam)
Oral exam: 25 points
Class work (practice): 25 points

To pass the written exam, a score of 25 points is required. Passing the written exam (by taking the preliminary exams or the final examination) is a prerequisite for taking the oral exam.
The final grade is based on scores attained in the written exam, oral exam and class work as follows:

50 points - 60 points = sufficient (2)
61 points - 74 points = good (3)
75 points - 87 points = very good (4)
88 points - 100 points = excellent (5)

1.10. Required reading (at the time of proposing the study programme)

1. Ibbotson, Mark: *Cambridge English for Engineering*, Cambridge University Press, Cambridge, 2008

1.11. Further reading (at the time of proposing the study programme)

1. Murphy, Raymond: *English Grammar in Use*, Cambridge University Press, Cambridge, 2004

1.12. Number of copies of required reading compared to the number of students currently taking the course

Title	*Number of copies*	*Number of students*
Ibbotson, Mark: *Cambridge English for Engineering*, Cambridge University Press, Cambridge, 2008	6	30

1.13. Methods of quality monitoring aimed at ensuring the acquisition of output knowledge, skills and competences
Conducting student surveys and evaluating the data in accordance with the results of the Student Survey Committee and the analyses of exam results in accordance with the data from the ISVU data warehouse.
1.14. Possibility of course instruction in a foreign language
English