| Funded by the European Union and the Council of Europe | EUROPEAN UNION | COUNCIL OF EUROPE CONSEIL DE L'EUROPE | Implemented by the Council of Europe |

Minimum elements of qualification standard

1. BASIC CHARACTERISTICS

1.1 Name(s) of the qualification (generic + subject specific)

Computer Science Engineer

1.2 Minimum volume

180 ECTS

1.3 Level

6

1.4 Entry routes

Completed four-year secondary education or equivalent.

2. COMPETENCIES / LEARNING OUTCOMES

2.1 Competencies at the level of qualification

Knowledge:

- Apply basic principles and methods of computer science to a wide range of applications
- Apply mathematical and scientific reasoning to a variety of computational problems
- Design, correctly implement and document solutions to significant computational problems

Application:

- Analyse and compare alternative solutions to computational problems
- Apply generally accepted principles to the synthesis and analysis of computer systems
- Apply advanced algorithmic and mathematical concepts to the design and analysis of software
- Design and implement software systems that meet the specified design and performance requirements

Preparations for lifelong learning:

- Work in teams to design and implement solutions to computational problems
- Communicate clearly and precisely, both verbally and in writing
- Think critically and creatively, both independently and with others
- Recognize the social and ethical responsibilities of professional work
- Keep up to date with and apply new knowledge in the field of computer science

2.2 Learning outcomes

Group	ECTS (minimum)
Mathematics	**16**
• Describe and explain the basic concepts of linear algebra, mathematical analysis, discrete mathematics, probability and statistics • Formulate and solve problems arising from mathematical situations, providing sound arguments for your view • Identify and recommend an appropriate model, structure and method of problem solving and statistical inference	
Theoretical Foundations of Computer Science	**10**
• Describe and explain the basic concepts of data structures and algorithms, theory of computation, formal methods and automata theory • Devise and analyse an algorithm or formal method for solving a specific problem • Solve problems by using standard algorithms for implementing and manipulating data structures	
Programming Paradigms and Programming Languages	**18**
• Describe and explain the basic programming paradigms: structured programming, object-oriented programming • Select an appropriate programming paradigm and programming language for problem solving • Independently create an application using various programming languages	
Operating Systems and System Programming	**6**
• Describe and explain operating system architecture • Describe and explain basic algorithms for: the process of managing memory and peripherals, file system organization, security • Select and use an appropriate operating system for various applications	
Computer Architecture and Computer Organization	**10**
• Describe and explain the basic principles of digital systems construction, their analyses and design • Describe and explain the basic elements of computer architecture and computer organization at the level of assembly • Design combinatorial and sequential circuits at the logical level • Develop a simple program at the assembly/machine level	
Computer Networks and Communication	**6**
• Describe network standards, concepts, topologies and media, network hardware, network security and various protocols • Explain the organization of the Internet and describe delivery and forwarding of IP packets	

Group	ECTS (minimum)
• Critically evaluate the factors affecting the performance of protocols and implement a simple, reliable protocol, compare and contrast fixed and dynamic allocation techniques, and identify the problem of multiple access	
Databases and Data Management	**8**
• Describe and explain the basic concepts, technologies and principles of the organization of relational databases, including the security aspects • Create a database at the conceptual, logical and physical level • Use declarative languages to create and manipulate a database	
Software Engineering and Software Development	**20**
• Explain and describe the basics of software engineering and software development: software processes and methodologies, software life-cycle processes, project management, software architectures and technologies, and the user interface • Individually apply different techniques for: reception and analysis of requests, design, implementation, testing and evolution of software solutions • Perform a specific task on a team developing a big software system • Be involved in the development of software systems for various applications (desktop, client/server, web, mobile, built-in, real-time applications)	
Computer and Information Security	**5**
• Describe and explain the basic elements of cryptography and key aspects of security (confidentiality, integrity, availability), the concepts of risk, threat, attack, authentication, authorization, access control • Individually apply, use, monitor and maintain various methods, techniques and tools for security of computer and information systems	
Foreign Languages	**5**
• Effectively use basic lexical and syntactic structures of a foreign language, both verbally and in written form • Demonstrate knowledge and skills in the use of a foreign language relevant to computer science, both verbally and in writing	
Final thesis/project	**6**
• Individually devise a solution (design, implement, document and present) to a simple engineering problem in the field of computer science, synthesizing the acquired knowledge, skills and abilities and using appropriate professional literature	

Minimum of 110 ECTS credits for the obligatory courses in computer science, electric engineering and mathematics.
Maximum ECTS credits for courses not related to computer science, electric engineering and mathematics: 30 ECTS credits

3. RELEVANCE

3.1 Labour market

- *Software Engineer (System Analyst, Software Designer, Programmer/Developer, Software Quality Controller, Maintenance)*
- *Information Systems Administrator*
- *Computer Systems Administrator*
- *Computer Systems Technical Support Engineer*
- *Teacher of Information Technology and Computing in elementary school (having previously qualified as a teacher by completing the necessary pedagogical and didactic training)*

3.2 Further education / progression

- *The second cycle of university education in the field of computing (computer science, software engineering, computer engineering, information technologies, information systems)*
- *The second cycle of university education in a related field*
- *Further professional advancement by gaining non-formal and informal qualifications*

3.3 Other needs

General need for the computerization of the society.

4. QUALITY ASSURANCE

4.1 Members of the Working Group

Academic community:

- Assoc. Prof. Samra Mujačić, University of Tuzla
- Assoc. Prof. Suad Kasapović, University of Tuzla
- Assoc. Prof. Samim Konjicija, University of Sarajevo
- Assist. Prof. Dražen Brđanin, University of Banja Luka
- Assist. Prof. Dragan Matić, University of Banja Luka
- Assist. Prof. Jasminka Hasić, International University of Sarajevo
- Assist. Prof. Zanin Vejzović, Džemal Bijedić University in Mostar
- Assist. Prof. Samir Lemeš, University of Zenica
- Assist. Prof. Nina Bijedić, Džemal Bijedić University in Mostar
- Sr. Teach. Assist. Amir Hajdar, University of Sarajevo

Employers in the ICT sector:

- Muhdin Mujačić, M.A., Procom Ltd., Tuzla
- Ferid Ajanović, B.Sc.E.E., AtlantBH Ltd., Sarajevo

Employers in the ICT sector:

- Muhdin Mujačić, M.A., Procom Ltd., Tuzla
- Ferid Ajanović, B.Sc.E.E., AtlantBH Ltd., Sarajevo