CONTENS

INFORMATION SHEET

Name of the university: J. Selye University

Name of the faculty: Faculty of Education

Code: KINF/AP/22 | **Name:** Computer architecture

Types, range and methods of educational activities:
Form of study: Lecture / Seminar / Practical
Recommended extent of course (in hours):
Per week: 2 / 0 / 1 **For the study period:** 26 / 0 / 13
Methods of study: present

Number of credits: 5

Recommended semester/trimester of study: 3.

Level of study: I.

Prerequisites:

Conditions for passing the subject:
The course is completed by a written examination, for which students can obtain 50% of the total number of points. During the semester, students will take two written examinations for which they can earn 30% of the total points and 20% of the points can be earned by completing a semester project.
In addition to contact teaching, students prepare for practicals, prepare for written examinations, work on the semester project and prepare for the examination.
A grade of A requires a minimum of 90 points, a grade of B requires a minimum of 80 points, a grade of C requires a minimum of 70 points, a grade of D requires a minimum of 60 points, and a grade of E requires a minimum of 50 points. Credit will not be awarded to a student who scores less than 50 points.

Results of education:
Knowledge:
Upon completion of the course, the student will:
- has theoretical knowledge of computer architecture,
- knows the principle of operation of individual computer elements,
- has a deeper knowledge of Von-Neumann architecture.

Skills:
Upon completion of the course, the student will:
- is able to apply the acquired knowledge in solving practical problems,
- can analyze and solve simple and more complex problems,
- is able to design various logic circuits and implement them.

Competences:
Upon completion of the course the student:
- can work efficiently and implement the acquired theoretical knowledge,
- shows independence in solving more complex problems.

Brief syllabus:
1. The meaning of the term computer architecture and the significance of its different parts.
2. Boolean algebra, logical elements.
3. Logic circuits - their design and implementation.
4. Building blocks of digital systems.

5. Computer memory, registers.
6. Data types, mathematical operations, operand types, instruction formats, addressing.
7. Arithmetic-logic unit, instruction execution (instruction cycle).
8. Bus types, principle of operation, serial and parallel buses (FSB, PCI, PCIe, HT, QPI), their characteristics, data transfers, transfer rates, character systems.
9. Programming approach to I/O, I/O operations performed in memory unit, DMA, I/O channel.
10. Interrupt system - IRQ.
11. Principles of operation of DRAM, SRAM, ROM and EEPROM.
12. Virtual computer - construction, principles of operation.
13. Intel, AMD, IBM and ARM processors, their architectures, evolution and development trends.

Literature:

1. CSERNY, L. : Mikroszámítógépek. Budapest : LSI Oktatóközpont, 2003. s. 330. ISBN 963 577 188 6.
2. SIMA D. – FOUNTAIN, T. – KACSUK, P.: Korszerű számítógép-architektúrák tervezési tér megközelítésben. Bicske : SZAK Kiadó, 1998, s. 809. ISBN 963 9131 09 1.
3. TANNENBAUM, A. S.: Számítógéparchitektúrák. Budapest : Panem Kiadó, 2001, s. 720. ISBN 963 545 282 9.
4. BENYÓ B.: Számítógép architektúrája. Szécsényi István Egyetem. Győr. 2006. http://jegyzet.sze.hu/letolt.php?dwn=1szamitogepekar.
5. Antal, I.: Informatikai algoritmusok I. ELTE. Budapest. 2005. http://compalg.inf.elte.hu/~tony/Elektronikus/Informatikai/Infalg1H.xml.

Language, knowledge of which is necessary to complete a course:
Hungarian or Slovak

Notes:
Student workload distribution:
50% - attendance at tutorials, preparation for examinations and exams,
50% - study of literature, preparation of term papers.

Evaluation of subjects
Total number of evaluated students: 0

A	B	C	D	E	FX
0.0	0.0	0.0	0.0	0.0	0.0

Teacher: Dr. habil. András Molnár, PhD., Ing. Ondrej Takáč, PhD.

Date of last update: 02.03.2022

Approved by:

INFORMATION SHEET

Name of the university: J. Selye University	
Name of the faculty: Faculty of Education	
Code: KINF/BS/22	**Name:** Bachelor seminar

Types, range and methods of educational activities:
Form of study: Lecture / Seminar / Practical
Recommended extent of course (in hours):
Per week: 0 / 1 / 0 **For the study period:** 0 / 13 / 0
Methods of study: present

Number of credits: 4

Recommended semester/trimester of study: 5.

Level of study: I.

Prerequisites:

Conditions for passing the subject:
Submission of a selected bibliography related to the topic of the final thesis and drafting of a part of the final thesis (10-12 pages).
Attendance at the seminar is compulsory. The student will prepare part of the final thesis and submit the bibliography.
The student must hand in the final paper to the tutor by the deadline.
If the student does not hand in the final part of the thesis within 7 days after the deadline, he/she will not receive the credits for the course.
The length of the part of the thesis to be handed in is determined by the instructor, the formal requirements are specified in the Rector's Directive 2/2021.
The essay must comply with the technical rules and ethics of citation.
The student's analytical-synthetic train of thought, the expression of personal opinion supported by theoretical knowledge, the definition of the problem and purpose of the essay, the way it is developed, the structure of the essay - logical structure and proportionate length of the individual sections, the work with literature and information sources (how they are selected and used), compliance with the basic formal requirements of the essay, compliance with the requirements for citation, the aesthetic and linguistic quality of the essay.

Results of education:
Knowledge:
The student can:
- list and explain the general requirements for the preparation of a final thesis, describe and characterise the content structure of the final thesis and its parts (introduction, main body, annexes),
- explain the concepts of phenomenon and fact, list and describe ways of investigating educational phenomena,
- to describe in more detail the basic methods of collecting and processing the data presented in the final report,
- identify the basic requirements for the author of a thesis, describe and describe the model, characteristics and structure of a thesis,
- list and explain the formal requirements for the final thesis,

- define the concept of an abstract, describe its structure, describe the characteristics of a quality abstract, list the most common mistakes in abstract preparation, distinguish an abstract from an annotation, an abstract, an abstract summary and an overview,
- explain the concepts of citation, quotation, paraphrase, compilation, plagiarism, distinguish between quotation and paraphrase, illustrate with examples the different citation and referencing techniques,
- define and interpret in their own words the basic concepts and motifs of the chosen subject area,
- know the basic terms used in the thesis,
- explain the terms used in an essay,
- construct (elaborate) the theoretical plane of the thesis, including all its important aspects,
- analyse and justify the conclusions of the thesis,
- critically analyse, re-evaluate and theorise the knowledge gained.

Skills:

The student can:
- write a draft of their own final thesis,
- explain the methodological rules for writing a final paper,
- define the main question and aim of the final thesis, formulating hypotheses where appropriate,
- plan a timetable for the preparation of the final thesis, including the content,
- work with literature (primary and secondary sources), search for information in library information databases,
- prepare the text of the final thesis, based on the knowledge acquired, by formulating ideas logically and accurately, creating a quality abstract, writing an introduction and conclusion, taking into account the criteria given,
- presenting the knowledge acquired in the field, recognising its complexity and drawing conclusions,
- apply knowledge of the ethics and techniques of citation and drafting,
- use correctly the various methods of citation and referencing and compile a bibliography correctly,
- create (develop) the practical aspects of the thesis, including all relevant aspects,
- analyse, synthesise and compare knowledge and propose solutions on this basis,
- draw conclusions through critical analysis and formulate their practical implications,
- critically analyse, reassess and apply the knowledge acquired in practice,
- present, discuss and argue their own knowledge in relation to the intended purpose of the thesis,
- be able to write a thesis on a chosen topic
- apply a critical approach,
- apply the principles of copyright, scientific ethics and relevant ISO and STN standards in the conduct of research

Competences:

The student:
- become aware of the importance of respecting academic ethics and the ethical implications for their own student and future teaching,
- act in accordance with the rules of good conduct,
- has mastered the basics of social appearance, and is dressed appropriately for the state examination,
- observes the ethical principles of summoning,
- Expresses his/her beliefs and opinions in a straightforward and honest manner, while accepting that the other party has the right to form his/her own opinion,
- bears and accepts the consequences of his/her own actions.

Brief syllabus:

1. Requirements for the final thesis in the SJE guidelines.
2. A concise description of the final thesis.
3. The importance of the final thesis
4. Selection of the topic for the final paper.
5. Preparation of a selected bibliography for the thesis.
6. Tasks and objectives of the final thesis.
7. Choosing the appropriate citation.
8. Content of the final thesis.
9. Formulating a strategy for the development of each section (chapter).
10. Working with textbooks and journals.
11. Use of the Internet and online publications.
12. preparing and carrying out the research, preparing the defence of the final thesis.

Literature:

1. ISO STN 690: Dokumentácia - Bibliografické odkazy – Obsah, forma a štruktúra. 1998.
2. KATUŠČÁK, D.: Ako písať záverečné a kvalifikačné práce. Nitra : Enigma, 2008, s. 164. ISBN 978 80 89132 45 4.
3. KIMLIČKA, Š.: Ako citovať a vytvárať zoznamy bibliografických odkazov : podľa noriem ISO 690 pre „klasické“ aj elektronické zdroje. Bratislava : Stimul, 2002, s. 82. ISBN 80-889-82-57-X.
4. Vnútorné predpisy UJS o záverečných prácach (zásady obsahovej náplne, štruktúra a formálna úprava záverečných prác). Dostupné v akademickom informačnom systéme univerzity: https://ais2.ujs.sk .

Language, knowledge of which is necessary to complete a course:
Hungarian or Slovak

Notes:
Percentages for each task:
Work done in seminars: 20 %.
Seminar paper: 80 %.
The student must complete at least 50 % of all assignments.

Evaluation of subjects
Total number of evaluated students: 0

A	B	C	D	E	FX
0.0	0.0	0.0	0.0	0.0	0.0

Teacher: prof. RNDr. Tibor Kmeť, CSc.

Date of last update: 02.03.2022

Approved by:

INFORMATION SHEET

Name of the university: J. Selye University	
Name of the faculty: Faculty of Education	
Code: KINF/ DBAU/22	**Name:** Database Application Development

Types, range and methods of educational activities:
Form of study: Lecture / Seminar / Practical
Recommended extent of course (in hours):
Per week: 0 / 0 / 2 **For the study period:** 0 / 0 / 26
Methods of study: present

Number of credits: 2

Recommended semester/trimester of study: 5.

Level of study: I.

Prerequisites:

Conditions for passing the subject:
During the semester, students solve practical problems for which they can get 50 points. At the end of the semester, students will complete a term project for which they may receive 50 points. A grade is at least 90 points, B grade is at least 80 points, C grade is at least 70 points, D grade is at least 60 points and E grade is at least 50 points. A student who scores less than 50 points cannot receive credit.

Results of education:
Educational results - knowledge:
After completing the subject, the student:
• After completing the subject, the student knows the basic principles of creating dynamic websites and applications,
• They will gain knowledge about the potential uses, advantages and disadvantages, as well as the use of professional terminology.
Learning outcomes - skills:
After completing the subject, the student:
• capable of creating a web application with a database connection
• to establish a connection between the client and the server.
• for application creation, login, user management,
• can design web application architecture independently,
• implements the server and client part, as well as the communication protocols between the components.
Educational results - competences:
After completing the subject, the student:
• After completing the course, the student is able to create interactive web applications.
• They can use their knowledge as a web developer, as a developer of a complete web solution (full-stack developer),
• Able to create web database solutions and develop administration pages or company websites, for the development of an information visualization system, for the display of aggregated information.

Brief syllabus:

1. Application creation process, users, requirements, specifications.
2. Design of actors, objects, data identification, individual-relationship diagram.
3. Relational database design, relational database management, database implementation.
4. Identification and representation of processes, determination of the necessary source data.
5. Translation of queries into the language of the database system.
6. Definition of input requirements, implementation, implementation with the help of forms.
7. Preparation of summaries, statistics, complex queries.
8. Making a report.
9. System debugging with additional functions of the database system.
10. Creating macros.
11. Making an offer.
12. Setting up and managing user access.
13. System integration.

Literature:

1. MILES, R. (2019). C# Programming. Yellow Book “Cheese” Edition 8.1.
2. NAKOV, S. et al (2013). FUNDAMENTALS OF COMPUTER PROGRAMMING.
3. WITH C#. Sofia ISBN 978-954-400-773-7.
4. BÁRTFAI, B. – BUDAVÁRI, O.: Adatbázis-kezelés. BBS-INFO Kft., 2002. - 138 s. - ISBN 9630034441.
5. RESCA, S. (2019). Hands-On RESTful Web Services with ASP.NET Core 3: Design production-ready, testable, and flexible RESTful APIs for web applications and microservices. ASIN¶:ƒB07MXLQR34
6. KOLOSZÁR, L. – TÓTH, Zs.: Adatbázis-kezelés. Nyugat-magyarországi Egyetem, 2012.
7. https://baranyilaszlozsolt.com/pciskola/Adatbazis_80.o.pdf

Language, knowledge of which is necessary to complete a course:
Hungarian or Slovak

Notes:
Distribution of student workload:
60% - attending classes, studying at home and preparing for exams,
40% - study of professional literature, practice of acquired knowledge, work on practicals assignments, preparation of semester work.

Evaluation of subjects
Total number of evaluated students: 0

A	B	C	D	E	FX
0.0	0.0	0.0	0.0	0.0	0.0

Teacher: Dr. habil. Attila Elemér Kiss, CSc., László Marák, PhD.

Date of last update: 02.03.2022

Approved by:

INFORMATION SHEET

Name of the university: J. Selye University	
Name of the faculty: Faculty of Education	
Code: KINF/DEI/22	**Name:** History of Informatics and ICT

Types, range and methods of educational activities:
Form of study: Lecture / Seminar / Practical
Recommended extent of course (in hours):
Per week: 0 / 2 / 0 **For the study period:** 0 / 26 / 0
Methods of study: present

Number of credits: 3

Recommended semester/trimester of study: 1.

Level of study: I.

Prerequisites:

Conditions for passing the subject:
Over the course of the semester, students study the history of computing and computer science from relevant book sources and the Internet. There are 2 quizzes during the semester that each student must take. The course ends with an exam. Grading is determined by the average of the 2 tests, each of which a student must pass at least 50% to be admitted to the exam.
The student is classified according to the average obtained in the tests (50%) and the exam (50%). A score of at least 90% is required for a grade of A, at least 80% for a grade of B, at least 70% for a grade of C, at least 60% for a grade of D, and at least 50% for a grade of E. Credit will not be awarded for a course if the student is not at least 50% successful.

Results of education:
Knowledge:
Students will know the tools and methods used to store and organize data in the development of computer science and computing. They know the history of computing, computers and computer science. Students know the basic principles of operation and basic concepts of not only computers but also their peripherals. They know the personalities who have contributed substantially to the development of computer science, computing and information and communication technologies not only on a global but also on a national scale.
Skills:
Upon successful completion of the course, students will be prepared to recognize and use the methods and tools they have learned and will be able to learn about new developments. They will be able to work independently, study the literature, present results and critically evaluate them.
Competences:
Upon successful completion of the course, students will be prepared to teach the methods and use the tools they have learned. They understand the links between the development of the underlying fields of computer science and computing itself.

Brief syllabus:
1. The development of counting from antiquity to the Middle Ages (tools used).
2. Demonstration of mechanical devices supporting the performance of the four basic operations (modern era).
3. Demonstration of mechanical devices supporting the performance of the four basic operations (recent era).

4. The transmission system invented by Charles Babbage.
5. Computing tools developed in the early 20th century.
6. Electromechanical devices used during World War II. 7. The first computer developed by John von Neumann.
8. The computer generation.
9. Punch plate, punch tape, method of entering all data. 10. Magnetic data storage, magnetic tape, HDD, optical data storage.
11. Development of processors, increasing computing capacity.
12. Data display modes (cathode ray tube monitors, needle printers, dot matrix and serial printers)

Literature:

1. STOFFA, V. a kol. Az informatika alapjai I. (Základy informatiky I.) 1. vyd. Komárno : Univerzita J. Selyeho, 2007. 369 s. ISBN 978-80-89234-29-5
2. STOFFOVÁ, V. a kol. Informatika, informačné technológie a výpočtová technika. Terminologický a výkladový slovník. Nitra : FPV UKF, 2001. 230 s. ISBN 80-8050-450-4.
3. ZELENÝ, J. – MANNOVÁ, B. Historie výpočetní techniky. Praha : Scientia, 2006. 184 s. ISBN 80-86960-04-8.
4. STOFFA, V.: Információs és kommunikációs technológiák a gyakorlatban I. Komárno 2008, Valeur, 321 str. ISBN 978 80 89234 69 1.
5. STOFFA, V.: Informačné a komunikačné technológie v praxi I. Komárno 2008, Valeur, 321 str. ISBN 978 80 89234 69 1.
6. KATONA GYULA Y. : A számítástudomány alapjai. Typotex Elektronikus Kiadó Kft., 2002, 192 s. ISBN 963 9326 24 0.
7. ZWETLER, O. – NEČAS, C. Dejiny věd a techniky I. Brno : MU, 1992. 97 s. ISBN 80-210-0401-0.
8. DLUHOŠ, J. – VALA, M. Vybrané kapitoly z dejín techniky. Ostrava : PdF OU, 1996. 61 s. ISBN 80-7042-112-6.
9. HOUDEK, F. Objevy a vynálezy tisíciletí. Praha : NLN, 2002. 456 s. ISBN 80-7106-475-0.
10. PATURI, F. Kronika techniky. Bratislava : Fortuna Print, 1993. 654 s. ISBN 80-7153-065-4.
11. REID, S. Vynálezy a objevy. Ostrava : Blesk, 1994. 128 s. ISBN 80-85606-52-6.
12. ZEITHAMMER, K. Vývoj techniky. 2. vyd. Praha : ČVUT, 1998. 274 s. ISBN 80-01-01725-7.

Language, knowledge of which is necessary to complete a course:
Hungarian or Slovak

Notes:
Distribution of students' workload:
40% - participation in classes, preparation for examinations and exams, 60% - study of literature, preparation of term papers.

Evaluation of subjects
Total number of evaluated students: 0

A	B	C	D	E	FX
0.0	0.0	0.0	0.0	0.0	0.0

Teacher: PaedDr. Márk Csóka, Dr. habil. Dr. Gábor Kiss, PhD.

Date of last update: 02.03.2022

Approved by:

INFORMATION SHEET

Name of the university: J. Selye University	
Name of the faculty: Faculty of Education	
Code: KINF/ DMI/22	**Name:** Discrete Mathematics for Computer Science
Types, range and methods of educational activities: **Form of study:** Lecture / Seminar / Practical **Recommended extent of course (in hours):** **Per week:** 2 / 1 / 0 **For the study period:** 26 / 13 / 0 **Methods of study:** present	
Number of credits: 5	
Recommended semester/trimester of study: 4.	
Level of study: I.	
Prerequisites:	
Conditions for passing the subject: During the semester, students work independently on homework in online system weBWorK, for which they can receive a total of 30 points. The course is finished by an exam where it is possible to obtain 70 points. For assessment A should be obtained at least 90 points, for assessment B at least 80 points, for assessment C at least 70 points, for assessment D at least 60 points, for assessment E at least 50 points.	
Results of education: Educational outcomes - skills: At the end of the course, students will know the basic mathematical tools necessary to complete the theoretical informatics courses. Educational outcomes - competencies: At the end of the course, students will know the relevant knowledge from disrete matehmatics for teaching computer science in primary and secondary school level.	
Brief syllabus: 1. Introduction to the Discrete Mathematics, Peano axioms, principle of Mathematical induction. 2. Set Theory – basic terms, set operations. 3. Relations and mappings, composition of mappings, equivalence relation. 4. Combinatorics – combinations and variations (with and without repetition). 5. Permutations (with and without repetition), combinatorial identities. 6. Binomial and Polynomial theorem, Inclusion–exclusion principle 7. Propositions and logical operations, tautologies. 8. Boolean algebra – binary Boolean functions, realization of Boolean functions by formulas, Equivalence of Boolean formulas, properties of elementary Boolean functions, principle of duality. 9. Canonic form of Boolean functions, full disjunctive normal form, Minimization of Boolean functions. 10. Divisibility, the fundamental theorem of arithmeti, euclidean algorithm 11.Properties of prime numbers, solving linear diophantine equations 12. Elementary graph theory	
Literature: 1. JABLONSKIJ, S. V.: Úvod do diskrétnej matematiky. Bratislava : Alfa, 1984., 278 s.	

2. JABLONSKIJ, S. V. a kol.: Diszkrét matematika a számítástudományban. Budapest : Műszaki
3. Könyvkiadó, 1980. 354 s. ISBN 978-963-1025-99-3
4. SZENDREI, Á.: Diszkrét matematika. Szeged : Polygon, 1998. 380 s. ISSN 1417-0590.
5. LOVÁSZ, L. – VESZTERGOMBI, K. – PELIKÁN, J.: Diszkrét matematika. Budapest :
6. Typotex, 2006. 292 s. ISBN 978-963-9664-02-9.
7. Csabina, Zoltánné: Matematika példatár 1.: Halmazelmélet, sorozatok
8. (https://dtk.tankonyvtar.hu/xmlui/bitstream/handle/123456789/8037/0027_MAT1.pdf?sequence=1)
9. Combinatorics: An Intuitive Introduction (https://www.probabilisticworld.com/intuitive-introduction-combinatorics/)

Language, knowledge of which is necessary to complete a course:
Hungarian or Slovak

Notes:
Students' load distribution:
40% - direct teaching, preparation for the exam
60% -study of teaching materials, work on homework

Evaluation of subjects
Total number of evaluated students: 0

A	B	C	D	E	FX
0.0	0.0	0.0	0.0	0.0	0.0

Teacher: doc. RNDr. József Bukor, PhD., RNDr. Štefan Gubo, PhD.

Date of last update: 02.03.2022

Approved by:

INFORMATION SHEET

Name of the university: J. Selye University	
Name of the faculty: Faculty of Education	
Code: KINF/ DS1/22	**Name:** Database systems

Types, range and methods of educational activities:
Form of study: Lecture / Seminar / Practical
Recommended extent of course (in hours):
Per week: 1 / 0 / 2 **For the study period:** 13 / 0 / 26
Methods of study: present

Number of credits: 4

Recommended semester/trimester of study: 5.

Level of study: I.

Prerequisites:

Conditions for passing the subject:
During the semester, students write two written papers, which are evaluated as a percentage. Students must achieve a score of at least 50% for both written papers in order to take the exam. During the semester, students work independently on a semester assignment or project (database management assignment). The combined exam consists of a written and oral part. To pass the exam, students must achieve at least 50% in the oral exam. The students are classified based on the obtained average, which includes the continuous performance of the semester, the work of the semester project and the result of the exam.
A grade is at least 90 points, B grade is at least 80 points, C grade is at least 70 points, D grade is at least 60 points and E grade is at least 50 points. A student who scores less than 50 points cannot receive credit.

Results of education:
Educational results - knowledge:
After completing the subject, the student:
• knows terms related to databases and their management.
• knows the characteristics of different database systems, the design of relational databases, the SQL language, the principles of creating forms and reports.
Learning outcomes - skills:
After completing the subject, the student:
• knows how to manage, use, query and create database systems.
Educational results - competences:
After completing the subject, the student:
• able to solve data management tasks using a database management system, formulate queries, prepare reports and forms.

Brief syllabus:
1. Basic concepts of database management, SQL language.
2. Relational data model, one-table SQL query.
3. Diagram of Entity-relationships, one-table complex queries in SQL.
4. Conversion of the diagram of relations into a relational database scheme, SQL query aggregation.
5. Normal forms, functional dependencies, multi-table SQL queries.

6. Decomposition into normal form, SQL queries using subqueries.
7. Creating databases, handling null values in SQL.
8. Updating data in SQL.
9. Management of access privilages and transactions.
10. Use of a database management system, data entry.
11. Use of a database management system, querying data.
12. Use of a database management system, creating forms.
13. Use of a database management system, creating reports.

Literature:

1. BALÁZS, P. – NÉMETH, G.: Adatbázisok. [Digitális Tankönyvtár]. Online dostupné: https://dtk.tankonyvtar.hu/xmlui/bitstream/handle/123456789/13212/adatbazisok.pdf
2. BÁRTFAI, B. – BUDAVÁRI, O.: Adatbázis-kezelés. BBS-INFO Kft., 2002. - 138 s. - ISBN 9630034441.
3. RESCA, S. (2019). Hands-On RESTful Web Services with ASP.NET Core 3: Design production-ready, testable, and flexible RESTful APIs for web applications and microservices. ASIN¶:ƒB07MXLQR34
4. KOLOSZÁR, L. – TÓTH, Zs.: Adatbázis-kezelés. Nyugat-magyarországi Egyetem, 2012.
5. https://baranyilaszlozsolt.com/pciskola/Adatbazis_80.o.pdf

Language, knowledge of which is necessary to complete a course:
Hungarian or Slovak

Notes:
Distribution of student workload:
50% - participation in lessons, preparation for background checks and exams,
50% - study of professional literature, practice of acquired knowledge, work on practicals assignments, preparation of semester work.

Evaluation of subjects
Total number of evaluated students: 0

A	B	C	D	E	FX
0.0	0.0	0.0	0.0	0.0	0.0

Teacher: Dr. habil. Attila Elemér Kiss, CSc., László Marák, PhD.

Date of last update: 02.03.2022

Approved by:

INFORMATION SHEET

Name of the university: J. Selye University	
Name of the faculty: Faculty of Education	
Code: KINF/ GED/22	**Name:** Graphics editors

Types, range and methods of educational activities:
Form of study: Lecture / Seminar / Practical
Recommended extent of course (in hours):
Per week: 0 / 0 / 2 **For the study period:** 0 / 0 / 26
Methods of study: present

Number of credits: 3

Recommended semester/trimester of study: 3.

Level of study: I.

Prerequisites:

Conditions for passing the subject:
During the semester, students' activity on the laboratory exercises is evaluated (with maximum score of 25 points). During the semester, students independently work on 3 semester projects (Paint.NET, Gimp and Inkscape), for which a total of 75 points can be obtained. The output of each project should be a tutorial. At the end of the semester, the students submit the finished tutorials in the form of a video together with documentations in text files, and these are evaluated. To receive grade A in the course, student must obtain at least 90 points, for grade B at least 80 points, for grade C at least 70 points, for grade D at least 60 points and for grade E at least 50 points. Credits will not be given to a student who obtain less than 50 points.

Results of education:
Knowledge:
After completing the subject, the student:
• has practical knowledge of raster graphics and vector graphics.
Skills:
• is able to use raster and vector graphics editors at an advanced level,
• is able to edit digital photography, work with tools, layers, filters, adjustments and effects,
• knows the rules of creation of documentations for semester projects.
Competencies:
• is able to work independently and efficiently with graphics editors.

Brief syllabus:
1. Basic terms of Computer graphics – vector and raster graphics, graphics file formats.
2. Raster graphics, overview of raster graphics editors (Paint.NET, Gimp).
3. Environment of the graphics editor: design area, grid, tool palette, status line, color palette, selection tools.
4. Work with drawing tools: paintbrush, pencil, eraser, magic wand, paint bucket, clone stamp, recolor, text tool. Drawing filled and unfilled rectangles (squares) and ellipses (circles).
5. Selection tools: rectangle select, ellipse select, operations on selections. Object selection and deselection, resize, translation, rotation, crop.
6. Work with text: inserting and editing text.

7. Work with layers: add and delete layers, layer selection, layer properties, changing the order of layers.
8. Raster image creation and processing.
9. Digital photo editing.
10. Work with adjustments and effects. Installing new plugins.
11. Vector graphics, overview of vector graphics editors (Inkscape).
12. Vector image creation and processing.
13. 3D graphics, overview of 3D graphics editors (Blender).

Literature:

1. TAKÁČ, O.: A számítógépes grafika. Komárno : Univerzita J. Selyeho, 2016. 370 s. ISBN 978-80-8122-182-8.
2. SZIRMAY-KALOS, L.: Számítógépes grafika. Budapest : ComputerBooks. 2003, 334 s. ISBN 978-963-6182-08-6.
3. NĚMEC, P.: GIMP 2.8 : Uživatelská příručka pro začínající grafiky. Brno : Computer Press, 2013. 272 s. ISBN 978-80-251-3815-1.
4. ŠIMČÍK, P.: Inkscape : Praktický průvodce tvorbou vektorové grafiky. Brno : Computer Press, 2013. 296 s. ISBN 978-80-251-3813-7.
5. BELAN, A.: Blender - malý úvod do 3D modelovania a animácie. Bratislava, 2008. Dostupné na: http://www.smnd.sk/anino/moje/blender/Blender.pdf
6. Paint.NET, https://forums.getpaint.net/
7. Gimp, https://www.gimp.org/tutorials/
8. Inkscape, https://inkscape.org/forums/

Language, knowledge of which is necessary to complete a course:
Hungarian or Slovak

Notes:
Distribution of the student's workload:
35% of the workload - direct teaching, preparation for laboratory exercises.
65% of the workload - studying the literature, practicing the acquired knowledge, work on the semester projects.

Evaluation of subjects
Total number of evaluated students: 0

A	B	C	D	E	FX
0.0	0.0	0.0	0.0	0.0	0.0

Teacher: RNDr. Štefan Gubo, PhD., PaedDr. Márk Csóka

Date of last update: 02.03.2022

Approved by:

INFORMATION SHEET

Name of the university: J. Selye University

Name of the faculty: Faculty of Education

Code: KINF/HW/22 | **Name:** Computer Hardware

Types, range and methods of educational activities:
Form of study: Lecture / Seminar / Practical
Recommended extent of course (in hours):
Per week: 0 / 0 / 2 **For the study period:** 0 / 0 / 26
Methods of study: present

Number of credits: 4

Recommended semester/trimester of study: 2.

Level of study: I.

Prerequisites:

Conditions for passing the subject:
There will be one written examination for 40 points during the semester. In seminars, students' active approach will also be assessed for 20 points. A grade of A requires at least 90% points, a grade of B requires at least 80% points, a grade of C requires at least 70% points, a grade of D requires at least 60% points, and a grade of E requires at least 50% points. Credit will not be awarded to a student who scores less than 50% on the written examination.

Results of education:
Knowledge:
After completing the course, the student knows the basic principles of logic circuits. The student knows the basic logic operators and logic modules. The student knows the internal components of computers and the operation of basic components such as the arithmetic and logic unit, control unit and memory. The student knows the basic differences between different architectures.
Skills:
After completing the course, the student is able to design simple logic circuits using simulation programs. The student is able to implement simple logic modules, memory modules, comparators and registers.
Competencies:
After completing the course, the student is proficient in the theory of logic circuits, is familiar with the internal components of computers, and is proficient in understanding the basic functions of the components. The student can recognize the differences between different architectures and is aware of the advantages and disadvantages of each architecture.

Brief syllabus:
1. Current, voltage, charged particles, electrical resistance, semiconductors and semiconductor components.
2. Fundamentals of logic circuits, diodes and transistors
3. Logic circuits. Binary logic operators.
4. Electrical implementation of logic circuits
5. Computer memory, D-Latch, Enabler, Register, Shift Register, Memory addressing
6. Computer bus, bus communication
7. Combination of logic gates, logic modules, addition module (ADD), comparison module (CMP)
8. Arithmetic and logic unit

9. Computer frequency, oscillator and timer, stepper
10. Control unit and instructions
11. Four basic types of instructions (arithmetic and logic instructions, instructions to manipulate the address of the current JMP instruction, comparison instructions, load and dump instructions)
12. Alternative architectures for general-purpose graphics processing units GPGPUs
13. Alternative architectures of user-programmable FPGA logic member arrays

Literature:

1. SCOTT, J. (2009). But how Do it Know?: The Basic Principles of Computers for Everyone. John C. Scott.
2. RAJEWSKI, J. (2017). Learning FPGAs (2017). O'Reilly Media, Inc. ISBN: 9781491965498.
3. SANDERS, J. - KANDROT, E. (2010), CUDA by Example: An Introduction to General-Purpose GPU Programming. Addison-Wesley Professional. ISBN: 9780132180160.

Language, knowledge of which is necessary to complete a course:
Hungarian or Slovak

Notes:
Student workload distribution:
50% - attendance at tutorials, preparation for revision and exam,
50% - studying literature, practicing the acquired knowledge.

Evaluation of subjects
Total number of evaluated students: 0

A	B	C	D	E	FX
0.0	0.0	0.0	0.0	0.0	0.0

Teacher: Dr. habil. András Molnár, PhD., Dr. habil. Sándor Szénási, PhD.

Date of last update: 02.03.2022

Approved by:

INFORMATION SHEET

Name of the university: J. Selye University	
Name of the faculty: Faculty of Education	
Code: KINF/OB/22	**Name:** Final thesis and its defence
Types, range and methods of educational activities: **Form of study:** **Recommended extent of course (in hours):** **Per week: For the study period:** **Methods of study:** present	
Number of credits: 8	
Recommended semester/trimester of study: 5., 6..	
Level of study: I.	
Prerequisites:	
Conditions for passing the subject: In preparing the final thesis, the student follows the instructions of his/her supervisor and the Rector's Directive regarding editing, registration, access and archiving of theses at J. Selye University. The recommended length of the bachelor thesis is 30 to 40 pages (54 000 to 72 000 characters including spaces). The deadline for the academic year is set in the academic calendar of the academic year. The originality of the thesis is evaluated in the central thesis register. The result of the originality check, a report on the originality of the thesis assessed. The originality check is a prerequisite for the defence. The submission of the thesis includes the conclusion of a licence agreement for the use of the digital reproduction of the thesis between the author and the Slovak Republic represented by the university. The final thesis shall be assessed by the thesis supervisor and a opponent, who shall draw up opinions according based on the established criteria. The thesis supervisor assesses in particular the fulfilment of the aim of the thesis, the degree of independence and initiative of the student in the elaboration of the topic, cooperation with the thesis supervisor, logical structure of the thesis, the adequacy of the methods used, the methodology, the professional level of the thesis, the depth and quality of processing of the topic, the contribution of the work, the possibility of using the results, the work with literature, the relevance of the sources used in relation to the topic and the aim of the thesis, the formal aspect of the thesis, spelling, stylistics and originality. The opponent assesses in particular the topicality and appropriateness of the topic of the thesis, the statement of the thesis and the content, the logical structure of the thesis, the continuity of the chapters, their proportionality, the appropriateness and suitability of the methods used, the methodology, the professional level of the thesis, the depth and quality of the treatment of the topic, the contribution of the thesis, the work with professional literature, the formal aspect, the spelling, the stylistics and originality. The State Examination Board will assess the originality of the thesis, the contribution of the student's work to the solution of the research problem, the student's independence, his/her ability to solve the research problem - from the search of literature sources, the determination of objectives, the choice of research methodology, the choice of the source of materials, through the implementation of the research, his ability to evaluate the results, discuss the results, summarize the results, present their significance for the educational process, etc. The ability to present the results is also evaluated, including answering questions related to the research process and the topic of the thesis, compliance with time limits, etc. The State Examination Board in a closed session will evaluate the course of the defence and decide on the award of the classification.	

In the classification, it comprehensively assesses the quality of the final thesis and its defence, taking into account the assessments and the course of the defence, and shall give the defence a single overall mark.
The final grade may be the same as in the evaluations, but it may also be better or worse, in depending on the conduct of the defence.
Final grade: A - 100 - 91%, B - 90 - 81%, C - 80 - 71%, D - 70 - 61%, E - 60 - 50%.
Credit will not be awarded to a student who fails to achieve 50%.
The decision on the result of the defence will be announced publicly by the chairman of the committee together with the result of the theoretical the theoretical part of the oral part of the state examination.

Results of education:
Knowledge:
- the student knows the structure of a scientific publication,
- the student can independently and creatively use professional sources,
- the student is able to analyse and evaluate the current state of the problem in his/her field,
- the student can adequately select research procedures and apply them functionally.
Skills:
- the bachelor thesis verifies mastery of the theoretical and practical foundations of the problem.
- the student should demonstrate the ability to work with domestic and foreign literature, The student should be able to select the relevant information for his/her topic, apply his/her skills in gathering, interpretation and processing of basic professional literature,
- the student has developed the skills of independent learning, which enables him/her to pursue further study,
- the student can collect and interpret relevant data (facts) within the field of study
and make informed decisions that also take into account social, scientific and ethical aspects,
- the student will be able to justify the ideas presented as well as to articulate them in a sophisticated manner in practical conclusions and recommendations,
- the student will be able to prepare a presentation of the results of the bachelor thesis,
- the student will be able to apply the principles of scientific integrity and ethics.
Competences:
- the student will be able to demonstrate his/her professional and scientific knowledge and skills in his/her field of study,
- the student is able to argue and methodically apply knowledge in theoretical or didactic and methodological contexts,
- the student is able to implement and synthesize the acquired knowledge in practice,
- the student is able to answer the questions of the supervisor and the opponent at the required level, to successfully defend the thesis.

Brief syllabus:
1. presentation of the thesis
2. presentation of the main points of the supervisor’s and the opponent’s verdict.
3. student's answers to the supervisor's and opponent's questions.
4. a professional discussion of the thesis with questions for the student.
The student's presentation of the thesis should include, in particular, the following points:
1. A brief justification of the choice of the topic, its topicality, practical contribution.
2. Clarification of the objectives and methods used in the elaboration of the thesis.
3. The main content problems of the thesis.
4. Conclusions and practical recommendations reached by the author of the thesis.

During the presentation, the student has at his/her disposal his/her own copy of the thesis or an electronic presentation. The speech is to be delivered independently, within 10 minutes timeframe. The student may use computer technology. The thesis is available to the committee before and during the defence.

Literature:
KATUŠČÁK, D. Ako písať vysokoškolské a kvalifikačné práce. Bratislava: Enigma, 2004. Aktuálna Smernica rektora o úprave, registrácii, sprístupnení a archivácii záverečných prác na Univerzite J. Selyeho – dostupné na https://www.ujs.sk/documents/Smernica_c.2-2021o_zaverecnych_pracach_.pdf

Language, knowledge of which is necessary to complete a course:
Hungarian or Slovak

Notes:
Undergraduate theses are supervised by the staff of the Department of Informatics. The defence of the bachelor's thesis takes place in front of an examination committee, whose members are appointed by the dean.

Evaluation of subjects
Total number of evaluated students: 0

A	B	C	D	E	FX
0.0	0.0	0.0	0.0	0.0	0.0

Teacher:

Date of last update: 04.03.2022

Approved by:

INFORMATION SHEET

Name of the university: J. Selye University

Name of the faculty: Faculty of Education

Code: KINF/OS/22	**Name:** Operating systems

Types, range and methods of educational activities:
Form of study: Lecture / Seminar / Practical
Recommended extent of course (in hours):
Per week: 1 / 0 / 2 **For the study period:** 13 / 0 / 26
Methods of study: present

Number of credits: 5

Recommended semester/trimester of study: 5.

Level of study: I.

Prerequisites:

Conditions for passing the subject:
Interim assessment during the semester: 50% of the total assessment. During the semester a written test with maximum score of 30 points will be held. During the semester, students independently work on a semester project, for its submission and presentation a total of 20 points can be obtained. Students must obtain at least 50% of the interim assessment to be allowed to take the exam.
Exam: 50% of the total assessment. The course is finished by written exam, on which 50 points can be obtained. To successfully pass the exam, it is necessary to obtain at least 50% of the exam evaluation.
The overall assessment consists of the sum of points from the interim assessment and the final exam. To receive grade A in the course, student must obtain at least 90 points, for grade B at least 80 points, for grade C at least 70 points, for grade D at least 60 points and for grade E at least 50 points. Credits will not be given to a student who obtain less than 50 points.

Results of education:
Knowledge:
After completing the subject, the student:
• has theoretical knowledge of the operating systems,
• knows scheduling algorithms,
• knows methods for detecting and handling deadlocks.
• knows algorithms for memory management,
• knows how file systems work,
• knows the operation and management of I/O devices.
Skills:
After completing the subject, the student:
• is able to apply the acquired knowledge to solve practical tasks,
• is able to apply scheduling algorithms to solve tasks,
• is able to apply methods for handling deadlocks to solve tasks,
• is able to apply memory allocation algorithms to solve tasks,
• knows the rules of creation of documentations for practical tasks.
Competencies:
After completing the subject, the student:
• is able to work independently and efficiently,

• has an active and responsible approach to completing tasks within the subject.

Brief syllabus:

1. Introduction to the Operating Systems, basic terms.
2. History and categorization of operating systems.
3. Programming interface and user interface.
4. Process management, process states, threads.
5. Interprocess communication and synchronization.
6. CPU-scheduling, scheduling algorithms.
7. Resource management - resource distribution, resource allocation chart, deadlock.
8. Resource management - detection and handling deadlocks.
9. Operating memory management.
10. Virtual memory management, paging and segmentation.
11. Files and file systems, directory structure, permissions.
12. I/O system, peripheral devices, device interface.
13. Storage management and mass-storage structure.

Literature:

1. ADAMIS, G. – KNAPP, G.: Operációs rendszerek. Budapest : LSI Oktatóközpont, 2002, 278 s. ISBN 963 577 251 3.
2. CSERNY, L.: Mikroszámítógépek. Budapest : LSI Oktatóközpont, 2003, 330 s. ISBN 963-577-188-6.
3. KÓCZY, A. – KONDOROSI, K. et al.: Operációs rendszerek mérnöki megközelítésben. Budapest : Panem Kiadó, 2000. 180 s. ISBN 978-963-545250-0.
4. HAMBÁLKOVÁ, V.: Operačné systémy. Bratislava : Univerzita Komenského, 2015. 105 s. Dostupné na: http://www.dcs.fmph.uniba.sk/~bernat/os.ls2021/os-new.pdf
5. TANENBAUM, A. S.: Modern Operating Systems. Upper Saddle River, NJ : Pearson Prentice-Hall, 2009. 1076 s. ISBN 978-0-13-813459-4.
6. SILBERSCHATZ, A.: Operating System Concepts. New York, NY : John Wiley & Sons, 2004. 956 s. ISBN 978-0-47-125060-0.

Language, knowledge of which is necessary to complete a course:

Hungarian or Slovak

Notes:

Distribution of the student's workload:
50% of the workload - direct teaching, preparation for the test and the exam.
50% of the workload - studying the literature, work on the semester project.

Evaluation of subjects

Total number of evaluated students: 0

A	B	C	D	E	FX
0.0	0.0	0.0	0.0	0.0	0.0

Teacher: prof. Dr. Annamária Várkonyiné Kóczy, DSc., Ing. Ondrej Takáč, PhD.

Date of last update: 02.03.2022

Approved by:

INFORMATION SHEET

Name of the university: J. Selye University	
Name of the faculty: Faculty of Education	
Code: KINF/ PER/22	**Name:** Computer peripherals
Types, range and methods of educational activities: **Form of study:** Lecture / Seminar / Practical **Recommended extent of course (in hours):** **Per week:** 1 / 0 / 1 **For the study period:** 13 / 0 / 13 **Methods of study:** present	
Number of credits: 3	
Recommended semester/trimester of study: 3.	
Level of study: I.	
Prerequisites:	
Conditions for passing the subject: During the semester, there will be 2 interim evaluations in the form of a written test based on the summary of knowledge from the exercise. The condition for admission to the exam is the achievement of at least 50% point evaluation from the tests. As part of the exam, theoretical knowledge of lecture topics is tested - in written form, 100 points each. The final assessment consists of an average of 50% of the results of the tests (exercises) and 50% of the results of the written part of the theoretical written exam. To get an A grade, you need to get an average of at least 90%, to get a B grade at least 80%, for a C grade at least 70%, for a D grade at least 60% and for an E grade at least 50% . A student will not receive an assessment if he does not achieve at least a 50% average.	
Results of education: Knowledge: After completing the subject, the student: • knows the principle of operation of computer peripherals, optical, electronic and mechanical solutions and their physical principles, • possesses knowledge and understanding of the classification of peripherals. Skills: After completing the subject, the student: • is able to apply the acquired knowledge when solving practical tasks, • has skill in applying the possibilities of using computer peripherals, • is able to decide on their merits and demerits when applied to solving a problem. Competencies: After completing the subject, the student: • applies professional terminology and can implement acquired theoretical knowledge, • shows independence in solving problems in the given topic.	
Brief syllabus: 1. System division of peripherals. Mathematical models of input and output peripherals, matrix model. 2. Input devices; contact, indoor and capacitive keyboards, their properties and electronic solutions. 3. GM, raster principle, optical and ultrasonic mouse, piezoelectric and magnetostrictive effect.	

4. Ways of sensing direction and rotation.
5. Tablets. One touch, multitouch, touch displays.
6. Scanners, additive color mixing. Stepper motor and step control.
7. Bar code readers and PLL.
8. VIDIKON camera, photoelectric effect.
9. Principle of operation and use of CCD sensors.
10. Output peripherals, 2D output.
11. Control of LCD and TFT monitors. MEMS and MOEMS projectors. CRT monitors and electronic solutions.
12. Principles of printing, matrix model, DPI. Inkjet printers, division and principles of operations. "H" bridge. Laser, LED and LCS printers. Thermal printers.
13. Writing data on magnetic, optical and magneto-optical carriers. Hard disk, GMR technology. CD-ROM, DVD, WORM (CD-W). Magneto-optical recording (MO, CD-R/W). Development trends.

Literature:

1. STOFFA, V. – CSÍZi, L. – SZŐKÖL, I. – TÓTH, K. – VÉGH, L.: Az informatika alapjai I. Selye János Egyetem, 2007. 268 s. ISBN 978-80-89234-29-5.
2. DÉSI, I. – NAGY, I.: Informatikai fogalmak kisszótára. Budapest : Korona, 2001. 248 s. ISBN 963 9376 16 7.
3. DANCSÓ, T.: Tudnivalók a számítógépről. Budapest : Nemzeti Tankönyvkiadó, 2002. 64 s.ISBN 963 19 3373 3.
4. SCHNEIDER, F.: Mi van a PC-ben = Felhasználói műszaki ismeretek. Gyula : APC-Stúdió BT., 1996. 66 s. ISBN 0008456.
5. ANTAL, P. - BÓTA, L.: Számítógépes konfigurációk. Dostupné online: https://dtk.tankonyvtar.hu/xmlui/handle/123456789/8671.

Language, knowledge of which is necessary to complete a course:
Hungarian or Slovak

Notes:
Student load distribution:
45% - participation in lessons, preparation for the exam,
55% - study of professional literature, practice of acquired knowledge.

Evaluation of subjects
Total number of evaluated students: 0

A	B	C	D	E	FX
0.0	0.0	0.0	0.0	0.0	0.0

Teacher: Dr. habil. András Molnár, PhD., Dr. habil. Sándor Szénási, PhD.

Date of last update: 02.03.2022

Approved by:

INFORMATION SHEET

Name of the university: J. Selye University

Name of the faculty: Faculty of Education

Code: KINF/PP/22 | **Name:** Programming propaedeutics

Types, range and methods of educational activities:
Form of study: Lecture / Seminar / Practical
Recommended extent of course (in hours):
Per week: 0 / 0 / 2 **For the study period:** 0 / 0 / 26
Methods of study: present

Number of credits: 3

Recommended semester/trimester of study: 2.

Level of study: I.

Prerequisites:

Conditions for passing the subject:
Students gain programming experience in the Imagine Logo microworld during the semester and create their own animations in the Logo Motion graphic environment. From the middle of the semester they independently solve the chosen programming task - semester project, the output of which is their own didactic project in Imagine.
During the semester are evaluated the tasks (small projects) submitted by the student (max. 4 points). The student can receive additional points if he continuously works on his own project (max. 2 points).
The subject ends with exam, where the final evaluation is based on the percentage of tasks completed by the student and the defense of his own project. The condition for admission to the oral exam is to achieve more than 50% of the points obtained from the programming tasks and the creation of own project.
The final evaluation of the subject is done as follows:
40% of the points from the programming task + 20% of the score for own project + 40% of the points for the oral part of own project defense.
90-100% required to achieve grade A; 80-89% for grade B; for grade C, 70-79%; 60-69% for D rating, 50-59% for E rating of the total score.

Results of education:
Knowledge:
After completing the subject, the student:
• knows the development strategies, methods and forms of students' programming knowledge within the discipline of their subject specialization;
• knows the basic principles of programming in the microworld of child-oriented programming languages;
• knows and can effectively apply the acquired programming knowledge;
• knows the basic principles of creating algorithms and knows the corresponding program structures.
Skills:
After completing the subject, the student:
• is able to analyze and solve problems using a programming language;
• is able to independently apply an algorithm to solve a specific problem;
• uses the language of the given microworld actively;

• has basic practical experience in selecting tasks;
• is able to plan and implement his own project.
Competencies:
After completing the subject, the student:
• shows a high degree of independence in creating programs (projects);
• knows how to work effectively independently;
• is characterized by creative thinking and independence;
• applies a creative IT way of thinking in his work;
• has an overview of the possibilities of teaching programming of different types and levels of schools - through child-oriented programming languages (microworlds), which enable the development of algorithmic thinking and the acquisition of programming experience in a playful form;
• has an active and responsible attitude towards the completion of subject tasks.

Brief syllabus:

1. Teaching programming at different levels and types of schools.
2. The place of child-oriented programming languages in the teaching process.
3. Turtle graphics - turtle, animated turtles.
4. Logo Motion - animation, timing, phases of turtles.
5. Basic control commands and elements of the Imagine program environment.
6. Data types - variables, text, buttons and working with them.
7. Commands to control the objects.
8. Subprograms - individual procedures.
9. Events of objects, reaction to events.
10. Conditions for managing the process.
11. Overlapping objects, testing objects
12. Multimedia possibilities of the Imagine environment.
13. Planning and implementing the own project - didactic application.

Literature:

1. CZAKÓOVÁ, K. – STOFFOVÁ, V. Kreativitás és az aktív tanulást támogató programkörnyezetek. In: Mikrovilág alkalmazások : Egyetemi tankönyv. 1. kiadás. Komárno : Univerzita J. Selyeho, 2016. s. 12-31. ISBN 978-80-8122-191-0.
2. CZAKÓOVÁ, K. Saját alkalmazás fejlesztése Imagine programkörnyezetben. In: Mikrovilág alkalmazások : Egyetemi tankönyv. 1. kiadás. Komárno : Univerzita J. Selyeho, 2016. s. 35-107. ISBN 978-80-8122-191-0STOFFOVÁ, V. – CZAKÓOVÁ, K.: Prostredie na učenie sa bádaním. In: Úvod do programovania v prostredí mikrosvetov : Vysokoškolská učebnica. Komárno : Univerzita J. Selyeho, 2016. 115 s. ISBN 978-80-8122-170-5.
3. STOFFOVÁ, V. – CZAKÓOVÁ, K.: Tvorba vlastných aplikácií v Imagine. In: Úvod do programovania v prostredí mikrosvetov : Vysokoškolská učebnica. Komárno : Univerzita J. Selyeho, 2016. 115 s. ISBN 978-80-8122-170-5.
5. STOFFA, V.: Algoritmizáció és programozás. (Algoritmizácia a programovanie) 1. kiadás, Komárom : Selye János Egyetem, Tanárképző Kar, 2005. 174 s. ISBN 80-969251-7-2.
6. TÓTH, P.: Gondolkodásfejlesztés az informatika oktatásban. Ligatura, 2004. 60 s. ISBN 9638611324xy.
7. VANKÓ, P.: Érdekes feladatok és játékok gyűjteménye mikrovilág környezetben. (Zbierka zaujímavých úloh a hier v prostredí Imagine). Komárno : Selye János Egyetem, 2010. DM.3784-PF.10.30A.6D. 43 s.
8. http://imagine.elte.hu/ [online]
9. http://imagine.infovek.sk [online]

10. http://logo.sulinet.hu/ [online]

Language, knowledge of which is necessary to complete a course:
Hungarian or Slovak

Notes:
Student workload:
40% - participation in classes, preparation for exams,
60% - studying literature, practicing acquired knowledge, working on programming tasks, preparing semester work (project).

Evaluation of subjects
Total number of evaluated students: 0

A	B	C	D	E	FX
0.0	0.0	0.0	0.0	0.0	0.0

Teacher: PaedDr. Krisztina Czakóová, PhD., Dr. habil. Attila Elemér Kiss, CSc.

Date of last update: 02.03.2022

Approved by:

INFORMATION SHEET

Name of the university: J. Selye University

Name of the faculty: Faculty of Education

Code: KINF/PR1/22 | **Name:** Programming 1

Types, range and methods of educational activities:
Form of study: Lecture / Seminar / Practical
Recommended extent of course (in hours):
Per week: 2 / 0 / 2 **For the study period:** 26 / 0 / 26
Methods of study: present

Number of credits: 5

Recommended semester/trimester of study: 1.

Level of study: I.

Prerequisites:

Conditions for passing the subject:
During the semester, students must pass at least two practical examinations, which are evaluated by percentage. Each student solves assigned programming tasks independently as part of homework and continuously submits a specified number of debugged programs that are evaluated. Students' activity during the practical classes is also monitored. Active students receive a certain bonus that is added to the student's score during the semester. Students from written practical examinations as well as submitted programs must obtain a minimum of 50% evaluation to be allowed to take the exam. The teacher who leads the practical classes will prepare the assessment of the students from the individual components of the ongoing training during the semester.
The exam is combined and consists of practical programming (solving assigned tasks) and verification of theoretical knowledge from algorithmization and programming. To be classified, students must be at least 50% successful in the exam. Students are classified according to the obtained average from the overall evaluation of the continuous training during the semester and the exam. To obtain an A classification, it is necessary to obtain an average of at least 90%, to obtain a B grade at least 80%, for a C grade at least 70%, for a D grade at least 60%, for an E grade at least 50%. Credits for the subject will not be awarded to a student who does not pass at least 50% of the individual parts.

Results of education:
Knowledges:
After completing the course, the students have the necessary knowledge to create simple C programs, they know what an algorithm is, what a sequence, selection and iteration are, what a structured flowchart is made of. They know the algorithms for calculating the sum and average of array elements, the algorithm for finding the elements of an array, the algorithm for finding the maximum and minimum elements of an array and their indices, the algorithm for mirroring an array, etc. They know the necessary data types, control structures, standard libraries and their frequently used functions, the syntax and semantics of the C programming language. They can transcribe the flowchart into program code.
Skills:
After completing the subject, students can analyze and solve simpler problems, they are able to develop algorithm as a series of logical steps, express it with a structured flowchart and rewrite it into a program code. They are proficient in the chosen programming environment and have basic

programming knowledge, effectively use standard control structures and elements of the chosen programming language.
Competencies:
After completing the course, students can independently solve simple programming tasks, create algorithms and simple programs in the C programming language.

Brief syllabus:

1. Basic properties of algorithms, their creation and expression. Verbal and graphical expression of algorithms. Basic algorithmic structures and their usage.
2. Creation of algorithms and algorithmic procedures to solve various problems and tasks.
3. Compilation of source code in C language. Preprocessor. Object files. Creating an executable program.
4. Structure of the program in the C programming language. Syntax and semantics.
5. Basic data types (int, float, double, char), strings (char[]). An internal representation of the standard data types of the programming language. Variables and constants.
6. Standard libraries of the C programming language (stdio.h, math.h, stdlib.h, time.h, limits.h, etc.). Standard input and output. Standard functions, their syntax and semantics.
7. Control structures: sequence (block), selection (condition, switch) and iterations (for loop, while loop, do while loop).
8. Functions. Creating functions without parameters and with parameters. Hierarchization of the structure of the program code. Global and local variables.
9. Static one-dimensional arrays (vectors). Indexes of array elements. Basic algorithms on arrays (sum and average of elements, finding array elements, determining minimum and maximum, determining minimum and maximum indices, merging and intersecting arrays, exchanging elements, sorting array elements, etc.).
10. Pointers. Representation of pointers in computer memory. Different types of pointers in C (void*, int*, double*). Dynamic memory allocation using pointers.
11. Pointers and arrays. Dynamically created arrays.
12. Complex data types - data structure. Statically and dynamically created arrays of structures.

Literature:

1. PROKOP, J.: Algoritmy v jazyku C a C++. 3. aktualizované vyd. Praha : Grada Publishing, 2015. 200 s. ISBN 978-80-247-5467-3.
2. PERRY, G., MILLER, D.: C Programming : Absolute Beginner's Guide . 3. vyd. Harlow : Pearson Education, 2014. 337 s. ISBN 978-0-7897-5198-0.
3. IVÁNYI, A.: Informatikai algoritmusok I. 1. vyd. Budapest : ELTE Eötvös Kiadó, 2004. 816 s. ISBN 963 463 664 0.
4. IVÁNYI, A.: Informatikai algoritmusok II. 1. vyd. Budapest : ELTE Eötvös Kiadó, 2005. 750s. ISBN 963 463 775 2.
5. KNUTH, D. E.: The Art of Computer Programming Vol. 1 : Fundamental Algorithms. 3. vyd. New York : Addison-Wesley, 2015. 652 s. ISBN 978-0-201-89683-1.
6. KNUTH, D. E.: The Art of Computer Programming Vol. 3 : Sorting and Searching. 2. vyd. New York : Addison-Wesley, 2015. 782 s. ISBN 978-0-201-89685-5.
7. SPRAUL, V. A.: Think like a programmer : An Introduction to Creative Problem Solving. 1. vyd. San Francisco : No Strach Press, 2012. - 233 s. - ISBN 978-1-59327-4245.
8. STOFFA, V.: Algoritmizáció és programozás I. Komárno : Selye János Egyetem, 2005. 174 s. ISBN 80-969251-7-2.
9. STOFFA, V., CZAKÓ, K., VÉGH, L.: Programozás a gyakorlatban : Algoritmizáció és programozás II. 1. vyd. Komárno : Selye János Egyetem, 2015. 124 s. ISBN 978-80-8122-146-0.

10. SIROKI, L: C programozás kezdőknek. https://sites.google.com/site/sirokilaszlo/programozas/c-programozas-kezdoknek
11. HOROVČÁK, P., PODLUBNÝ, I.: Úvod do programovania v jazyku C. http://people.tuke.sk/igor.podlubny/C/index.htm
12. KRIVÁ, Z.: Základy programovania v jazyku C. Bratislava : STU, 2020. https://www.svf.stuba.sk/buxus/docs/dokumenty/skripta/Kriva_Z._-_ZAKLADY_PROGRAMOVANIA_V_JAZYKU_C.pdf
13. C Tutorial. https://www.tutorialspoint.com/cprogramming/index.htm
14. Learn C Programming. https://www.programiz.com/c-programming
15. VÉGH, L.: Interaktív animációk az algoritmusok és a programozás tanítására. https://anim.ide.sk/

Language, knowledge of which is necessary to complete a course:
Hungarian or Slovak

Notes:
Students‘ load distribution:
40% - participation in classes, preparation for exams,
60% - studying literature, practicing acquired knowledge, completing programming tasks.

Evaluation of subjects
Total number of evaluated students: 0

A	B	C	D	E	FX
0.0	0.0	0.0	0.0	0.0	0.0

Teacher: prof. József Zoltán Kató, DSc., PaedDr. Ladislav Végh, PhD.

Date of last update: 02.03.2022

Approved by:

INFORMATION SHEET

Name of the university: J. Selye University

Name of the faculty: Faculty of Education

Code: KINF/PR2/22 | **Name:** Programming 2

Types, range and methods of educational activities:
Form of study: Lecture / Seminar / Practical
Recommended extent of course (in hours):
Per week: 2 / 0 / 2 **For the study period:** 26 / 0 / 26
Methods of study: present

Number of credits: 5

Recommended semester/trimester of study: 2.

Level of study: I.

Prerequisites:

Conditions for passing the subject:
During the semester, students must pass at least two practical examinations, which are evaluated by percentage. Each student solves assigned programming tasks independently as part of homework and continuously submits a specified number of debugged programs that are evaluated. Students' activity during the practical classes is also monitored. Active students receive a certain bonus that is added to the student's score during the semester. Students from written practical examinations as well as submitted programs must obtain a minimum of 50% evaluation to be allowed to take the exam. The teacher who leads the practical classes will prepare the assessment of the students from the individual components of the ongoing training during the semester.
The exam is combined and consists of practical programming (solving assigned tasks) and verification of theoretical knowledge from algorithmization and programming. To be classified, students must be at least 50% successful in the exam. Students are classified according to the obtained average from the overall evaluation of the continuous training during the semester and the exam. To obtain an A classification, it is necessary to obtain an average of at least 90%, to obtain a B grade at least 80%, for a C grade at least 70%, for a D grade at least 60%, for an E grade at least 50%. Credits for the subject will not be awarded to a student who does not pass at least 50% of the individual parts.

Results of education:
Knowledges:
After completing the course, students have the knowledge needed to create more complex programs in C programming language. They know various sorting algorithms and the differences between them, basic algorithms for working with matrices and multidimensional arrays. They know various programming techniques: recursion and backtracking. They know the method of working with files and know the necessary functions for this. They have knowledge of dynamic data structures: linear lists, cyclic lists, tree structures. They know different methods of creating software products.
Skills:
After completing the course, students can analyze and solve more complex problems, they know how to compile a solution algorithm even for a more complex problem and rewrite the algorithm into program code in the C language. They skillfully use the chosen programming environment

and have more advanced programming skills, the use of recursion and backtracking, the effective use of dynamic data structures and different programming methods.
Competencies:
After completing the course, students demonstrate independence in solving more complex programming problems, creating algorithms and more complex programs in the C programming language.

Brief syllabus:

1. Sorting as a suitable example for searching for an efficient algorithm: simple exchange sort, bubblesort, insertion sort, selection sort. Time computational complexity of sorting algorithms.
2. Programming technique: recursion. Solving simple recursion problems. Solving the Tower of Hanoi problem with recursion.
3. Sorting algorithms using recursion: quicksort, mergesort. Time computational complexity of sorting algorithms using recursion.
4. Two- and multidimensional fields. Basic algorithms on matrices (sum and average of elements, finding elements of a matrix, determining the minimum and maximum, determining the minimum and maximum indices, exchanging elements, sorting the matrix, working with rows and columns of the matrix, etc.). Using multidimensional arrays.
5. Programming technique: backtracking. The problem of eight queens.
6. Solving other backtracking tasks: Finding a path in a labyrinth, Moving a horse on a chessboard.
7. File as a useful tool for transferring data between programs and their environment. File structure, declaration, file type, file access, file operations. Standard functions for working with files. Methods of working with files.
8. Dynamic data types and structures: Concept of dynamic variable, their representation in computer memory. Examples of dynamic data structures: linear list, stack, queue, and their use in programming.
9. Implementation of standardized data structures (linear one-way list, linear two-way list, cyclic lists, tree structures, network structures). Using appropriate data structures to simplify problem solving.
10. Creation of software products. Top-down method, bottom-up method, functional programming, modular programming.
11. Creation of program systems. The procedure for creating a program to solve a problem: problem analysis, problem reformulation, decomposition, etc. Methods of creating program projects and their characteristics. Cooperation and management of the work of the programming team.
12. Solving complex programming problems and assignments.

Literature:

1. PROKOP, J.: Algoritmy v jazyku C a C++. 3. aktualizované vyd. Praha : Grada Publishing, 2015. 200 s. ISBN 978-80-247-5467-3.
2. PERRY, G., MILLER, D.: C Programming : Absolute Beginner's Guide . 3. vyd. Harlow : Pearson Education, 2014. 337 s. ISBN 978-0-7897-5198-0.
3. IVÁNYI, A.: Informatikai algoritmusok I. 1. vyd. Budapest : ELTE Eötvös Kiadó, 2004. 816 s. ISBN 963 463 664 0.
4. IVÁNYI, A.: Informatikai algoritmusok II. 1. vyd. Budapest : ELTE Eötvös Kiadó, 2005. 750s. ISBN 963 463 775 2.
5. KNUTH, D. E.: The Art of Computer Programming Vol. 1 : Fundamental Algorithms. 3. vyd. New York : Addison-Wesley, 2015. 652 s. ISBN 978-0-201-89683-1.
6. KNUTH, D. E.: The Art of Computer Programming Vol. 3 : Sorting and Searching. 2. vyd. New York : Addison-Wesley, 2015. 782 s. ISBN 978-0-201-89685-5.

7. SPRAUL, V. A.: Think like a programmer : An Introduction to Creative Problem Solving. 1. vyd. San Francisco : No Strach Press, 2012. - 233 s. - ISBN 978-1-59327-4245.
8. STOFFA, V.: Algoritmizáció és programozás I. Komárno : Selye János Egyetem, 2005. 174 s. ISBN 80-969251-7-2.
9. STOFFA, V., CZAKÓ, K., VÉGH, L.: Programozás a gyakorlatban : Algoritmizáció és programozás II. 1. vyd. Komárno : Selye János Egyetem, 2015. 124 s. ISBN 978-80-8122-146-0.
10. SIROKI, L: C programozás kezdőknek. https://sites.google.com/site/sirokilaszlo/programozas/c-programozas-kezdoknek
11. HOROVČÁK, P., PODLUBNÝ, I.: Úvod do programovania v jazyku C. http://people.tuke.sk/igor.podlubny/C/index.htm
12. KRIVÁ, Z.: Základy programovania v jazyku C. Bratislava : STU, 2020. https://www.svf.stuba.sk/buxus/docs/dokumenty/skripta/Kriva_Z._-_ZAKLADY_PROGRAMOVANIA_V_JAZYKU_C.pdf
13. C Tutorial. https://www.tutorialspoint.com/cprogramming/index.htm
14. Learn C Programming. https://www.programiz.com/c-programming
15. VÉGH, L.: Interaktív animációk az algoritmusok és a programozás tanítására. https://anim.ide.sk/

Language, knowledge of which is necessary to complete a course:
Hungarian or Slovak

Notes:
Students' load distribution:
40% - participation in classes, preparation for exams,
60% - studying literature, practicing acquired knowledge, completing programming tasks.

Evaluation of subjects
Total number of evaluated students: 0

A	B	C	D	E	FX
0.0	0.0	0.0	0.0	0.0	0.0

Teacher: prof. József Zoltán Kató, DSc., PaedDr. Ladislav Végh, PhD.

Date of last update: 02.03.2022

Approved by:

INFORMATION SHEET

Name of the university: J. Selye University	
Name of the faculty: Faculty of Education	
Code: KINF/PR3/22	**Name:** Programming 3
Types, range and methods of educational activities: **Form of study:** Lecture / Seminar / Practical **Recommended extent of course (in hours):** **Per week:** 1 / 0 / 2 **For the study period:** 13 / 0 / 26 **Methods of study:** present	
Number of credits: 4	
Recommended semester/trimester of study: 3.	
Level of study: I.	
Prerequisites:	
Conditions for passing the subject: During the semester, each student must create one project (a complex application in the C# language under the Windows operating system) as part of their homework, which they must submit at the end of the semester. Each student must receive a minimum of 50% assessment in order to be allowed to take the exam. Students' activity during exercises is also monitored. Active students receive a certain bonus, which is added to the student's evaluation during the semester. The exam is combined and consists of practical programming (solving the assigned task) and verification of theoretical knowledge from object-oriented programming. In order to be classified, students must be at least 50% successful in the exam. Students are classified according to the obtained average from the overall evaluation of the continuous training during the semester (submitted project + activity during the exercises) and the exam. To obtain an A classification, it is necessary to obtain an average of at least 90%, to obtain a B grade at least 80%, for a C grade at least 70%, for a D grade at least 60%, for an E grade at least 50% . Credits for the subject will not be awarded to a student who does not pass at least 50% of the individual parts.	
Results of education: Knowledge: After completing the course, students have the necessary knowledge to create an application with a graphical interface in the C# language. They know the object-oriented programming paradigm, different concepts of object-oriented programming, some standard classes of the C# language. They know the method of creating a complex application with a graphical interface under the Windows operating system. Skills: After completing the course, students can analyze and solve more complex problems, they can create a complex application with a graphical interface in the C# language. They skillfully use the chosen programming environment and have more advanced skills in programming applications with a graphical interface under the Windows operating system. Competencies: After completing the course, students demonstrate independence in solving more complex programming tasks, creating complex applications with a graphical interface under the Windows operating system in the C# programming language.	

Brief syllabus:

1. Programming under the Windows operating system, overview of programming languages, visual, event-driven programming.
2. Programming in the C# language. Overview of C# data types and structures. Value data types (struct) and reference data types (class). Converting data, using the Convert static class. Basic C# components and events (Label, Button, TextBox, CheckBox, RadioButton, ListBox, etc.), component properties and events.
3. Object-oriented programming (OOP). Encapsulation, polymorphism, inheritance, class and object. Data (attributes) and methods. Constructor. Access to data and methods, visibility modifiers (public, private, protected).
4. Inheritance, polymorphism. Static and dynamic type, static and dynamic binding (early binding, late binding). Class hierarchy, object in C#. Compatibility and class conversion.
5. Overloaded methods, overloaded constructor. Examples of their use in C#.
6. Abstract class, abstract methods. Examples of using abstract classes.
7. Static classes, static methods and static data. Examples of using static classes.
8. Standard dialog boxes and their use in C# (ColorDialog, FontDialog, OpenFileDialog, SaveFileDialog).
9. Working with files. Streams in C#, classes Stream, BufferedStream, and FileStream. Reading and writing text files in C#, using methods File.ReadAllText, File.WriteAllText, and classes StreamReader, StreamWriter.
10. Graphics, drawing. Paint Event and Invalidate Method in C#. Classes and structures used in drawing: to define coordinates (Point, Rectangle), line and fill color (Pen, SolidBrush), drawing using methods of the Graphics class (DrawLine, DrawImage, DrawRectangle, FillRectangle, DrawEllipse, FillEllipse).
11. Comparison of usability of structure (struct) and class (class) in C# language. Defining and using the enumeration type (enum), creating and using interfaces (interface).
12. Genericity (generics) and generic collections in the C# language: classes List, LinkedList, Dictionary, SortedList, HashSet, SortedSet, Queue, Stack.
13. Exceptions, exception classes in C# (Exception, FormatException, IOException, FileNotFoundException). Handling exceptions with the try-catch-finally command, creating exceptions with the throw keyword. Defining and using custom exception classes.

Literature:

1. ANDERSON, T.: C# in Easy Steps. 1. vyd. Southam : Computer Step, 2004. 192 s. ISBN 1-84078-150-5.
2. HANÁK, J.: C# praktické príklady. 1. vyd. 290 s. ISBN 80-247-0988-0.
3. ARCHER, T.: Myslíme v jazyku C# = Knihovna programátora. 2. vyd. Praha : Grada Publishing, 2002. 308 s. ISBN 80-247-0301-7.
4. PETZOLD, CH.: Programování Microsoft Windows v jazyce C#. 1. vyd. Praha : SoftPress, 2003. 600 s. ISBN 80-86497-54-2.
5. KOTSIS, D., SZÉNÁSI, S.: Többnyelvű programozástechnika : Object Pascal, C++, C#, Java. 1. vyd. Budapest : Panem Könyvkiadó Kft., 2007. 580 s. ISBN 978 9 635454 72 3.
6. ILLÉS, Z. Programozás C# nyelven. Budapest, 2005. http://compalg.inf.elte.hu/~tony/Informatikai-Konyvtar/09-Programozas%20C-sharp%20nyelven/Programozas-Csharp-nyelven-Konyv.pdf.
7. KOVÁCS, E., RADVÁNYI, T., KIRÁLY, R., HERNYÁK, Z.: C# feladatgyűjtemény. 2011. https://dtk.tankonyvtar.hu/xmlui/bitstream/handle/123456789/8447/0046_csharp_feladatgyujtemeny.pdf.
8. C# Tutorial. https://www.tutorialspoint.com/csharp/index.htm.

9. C# Tutorials. https://www.tutorialsteacher.com/csharp.

Language, knowledge of which is necessary to complete a course:
Hungarian or Slovak

Notes:
Students' load distribution:
40% - participation in lessons, preparation for the exam,
60% - study of professional literature, practice of acquired knowledge, work on programming tasks, preparation of semester work.

Evaluation of subjects
Total number of evaluated students: 0

A	B	C	D	E	FX
0.0	0.0	0.0	0.0	0.0	0.0

Teacher: Dr. habil. Sándor Szénási, PhD., PaedDr. Ladislav Végh, PhD.

Date of last update: 02.03.2022

Approved by:

INFORMATION SHEET

Name of the university: J. Selye University	
Name of the faculty: Faculty of Education	
Code: KINF/PR4/22	**Name:** Programming 4
Types, range and methods of educational activities: **Form of study:** Lecture / Seminar / Practical **Recommended extent of course (in hours):** **Per week:** 0 / 0 / 2 **For the study period:** 0 / 0 / 26 **Methods of study:** present	
Number of credits: 3	
Recommended semester/trimester of study: 4.	
Level of study: I.	
Prerequisites:	
Conditions for passing the subject: During the semester, students have to pass three written tests on the main topics, which are evaluated by percentage. Students should prepare for the examinations at home by solving practical assignments. Students must obtain a minimum of 50% in each written examination to be allowed to take the exam. The overall evaluation of the continuous training of each student is calculated from the averages of three written examinations. The exam consists of practical programming (solving the given task). In order to be classified, students must be at least 50% successful in the exam. Students are classified according to the obtained average from the overall evaluation of the continuous training during the semester and the exam. To obtain an A classification, it is necessary to obtain an average of at least 90%, to obtain a B grade at least 80%, for a C grade at least 70%, for a D grade at least 60%, for an E grade at least 50% . Credits for the subject will not be awarded to a student who does not pass at least 50% of the individual parts.	
Results of education: Knowledge: After completing the course, students have knowledge of object-oriented programming, they have knowledge of creating programs in the Java language. They know what generic types are and know the generic collections of the Java language. Skills: After completing the course, students can create and use classes and objects, use interfaces, events, generic types in Java. They skillfully use the chosen programming environment and have skills in programming in the Java language, effective use of generic collections of the Java language. Competences: After completing the subject, students demonstrate independence in solving complex programming tasks, creating object-oriented programming code in Java.	
Brief syllabus: 1. Basics of the Java programming language: data types, control structures, syntax and semantics of the language. Getting to know the development environment. 2. Using strings, using single and multidimensional arrays in Java, using the Random class to generate random numbers.	

3. Solving simpler assignments to practice programming in the Java language.
4. Classes and objects, attributes and methods, constructor, visibility modifiers in Java.
5. Theory of class inheritance, its use, inheritance in the Java language.
6. Exceptions, Use of Exceptions in Java.
7. Interfaces, creating and using interfaces.
8. Polymorphism in the Java language.
9. Java Stream I/O. Working with files.
10. Generic types, creation and use of generic classes.
11. Java Collections, possibilities of their use.
12. Solving complex programming tasks in the Java language.

Literature:

1. CADENHEAD, R.: Tanuljuk meg a java programozási nyelvet 24 óra alatt. 1. vyd. Budapest : Kispapu, 2006. 527 s. ISBN 963 963707 6.
2. BURD, B.: Java. 2. vyd. Budapest : Panem Könyvek, 2017. - 503 s. - ISBN 978-615-5186-52-3.
3. KOTSIS D., SZÉNÁSI S.: Többnyelvű programozástechnika : Object Pascal, C++, C#, Java. 1. vyd. Budapest : Panem Könyvkiadó Kft., 2007. 580 s. ISBN 978 9 635454.
4. MCGRATH, M.: JAVA. 5. vyd. Leamington : In Easy Steps, 2014. 192 s. ISBN 978-1-84078-621-7.
5. SZÉNÁSI, S.: Java programozási nyelv oktatása C# alapokon. Informatika a felsőoktatásban 2008, Debrecen, Magyarország, 2008, pp. 1-7.

Language, knowledge of which is necessary to complete a course:

Hungarian or Slovak

Notes:

Students' load distribution:
50% - participation in lessons, preparation for and exams,
50% - study of professional literature, practice of acquired knowledge, work on programming tasks, preparation of semester work.

Evaluation of subjects

Total number of evaluated students: 0

A	B	C	D	E	FX
0.0	0.0	0.0	0.0	0.0	0.0

Teacher: Dr. habil. Sándor Szénási, PhD., PaedDr. Ladislav Végh, PhD.

Date of last update: 02.03.2022

Approved by:

INFORMATION SHEET

Name of the university: J. Selye University	
Name of the faculty: Faculty of Education	
Code: KINF/PS/22	**Name:** Computer networks

Types, range and methods of educational activities:
Form of study: Lecture / Seminar / Practical
Recommended extent of course (in hours):
Per week: 2 / 0 / 1 **For the study period:** 26 / 0 / 13
Methods of study: present

Number of credits: 4

Recommended semester/trimester of study: 6.

Level of study: I.

Prerequisites:

Conditions for passing the subject:
The course is completed by a written examination, for which students can obtain 50% of the total number of points. During the semester, students will take two written examinations for which they may earn 50% of the total points.
In addition to contact teaching, students prepare for practicals, prepare for written quizzes, and prepare for the exam.
A minimum of 90 points is required for a grade of A, a minimum of 80 points for a grade of B, a minimum of 70 points for a grade of C, a minimum of 60 points for a grade of D, and a minimum of 50 points for a grade of E. Credit will not be awarded to a student who scores less than 50 points.

Results of education:
Knowledge:
Upon completion of the course, the student will:
- has theoretical knowledge of computer networks,
- The student has a good knowledge of network models and transmission technologies,
- knows the individual network competences,
- Knows basic network protocols and addressing.

Skills:
Upon completion of the course, the student will:
- Is able to design and implement a local area network including configuration,
- is able to combine different network components and standards,
- is able to independently implement network protocols.

Competencies:
Upon completion of the course the student will:
- can work effectively and implement the acquired theoretical knowledge,
- shows independence in solving more complex problems.

Brief syllabus:
1. Concept of network, basic parts of a network.
2. Reasons for the introduction of computer networks and the resulting basic network services.
3. Basic types of computer networks (typology, topology, architecture).
4. LANs, (MAN, WAN).

5. Basic components of computer networks.
6. Internet, origin and development.
7. Methods of access.
8. Network transmission technologies.
9. ISO-OSI model.
10. TCP/IP protocol.
11. Internet applications and protocols.
12. IP address theory, domain addresses, content creation.
13. Basics of security in computer networks.

Literature:

1. ROUBEL, P.: Hardware pro úplné začátečníky. Brno : Computer Press, 2003. ISBN 8072267302
2. SOSINKY, B.: Počítačové sítě : Vše, co potřebujete vědět o správě sítí. Brno : Computer Press, 2010. ISBN 978-80-251-3363-7
3. STOFFOVÁ, V.: Az informatika alapjai II - A számítógépes hálózatok. (Základy informatiky II – Počítačové siete.). 1. vyd. Komárno : Univerzita J. Selyeho, 2010, s. 140. ISBN 978-80-89234
4. CSIZMADIA, J.: Számítógépes hálózatok architektúrája - Elektronikus tankönyv. Komárno. Selye János Egyetem, 2009.
5. GYÁNYI, S.: Informatika 2. Óbudai Egyetem. 2014. http://dtk.tankonyvtar.hu/xmlui/handle/123456789/12567.

Language, knowledge of which is necessary to complete a course:
Hungarian or Slovak

Notes:
Student workload distribution:
50% - attendance at tutorials, preparation for examinations and exams,
50% - studying literature, practicing the acquired knowledge.

Evaluation of subjects
Total number of evaluated students: 0

A	B	C	D	E	FX
0.0	0.0	0.0	0.0	0.0	0.0

Teacher: Dr. habil. Dr. Gábor Kiss, PhD., Ing. Ondrej Takáč, PhD.

Date of last update: 02.03.2022

Approved by:

INFORMATION SHEET

Name of the university: J. Selye University	
Name of the faculty: Faculty of Education	
Code: KINF/ ROB1/22	**Name:** Robotics
Types, range and methods of educational activities: **Form of study:** Lecture / Seminar / Practical **Recommended extent of course (in hours):** **Per week:** 0 / 0 / 2 **For the study period:** 0 / 0 / 26 **Methods of study:** present	
Number of credits: 2	
Recommended semester/trimester of study: 6.	
Level of study: I.	
Prerequisites:	
Conditions for passing the subject: During the semester, students pass two written examinations, for which they can receive 100% of the total number of points. In addition to contact teaching, students prepare for exercises, prepare for written examinations and the exam. It is necessary to get at least 90% of points for A rating, at least 80% of points for B rating, at least 70% of points for C rating, at least 60% of points for D rating and at least 50% of points for E rating. Credits will not be granted to a student who has not collected 50% of points at the end of the semester.	
Results of education: Knowledge: Upon completion of the course, the student will: - knows the professional terminology, - knows the basic principles of mobile and stationary robots, - knows the principles of navigation of mobile robots, - knows the principles of positioning systems, - knows the individual functional and structural parts of robots, - has a deeper knowledge of autonomous systems and their use in a wide range of applications. Skills: Upon completion of the course, the student will: - Can design mobile or stationary robotic systems, - can design and implement multisensor systems, - can mathematically evaluate navigation signals, - can analyze and solve basic problems of stationary or mobile robots, - Can navigate robotic systems and use them in specific applications. Competencies: Upon completion of the course, the student will: - Can work efficiently and implement the acquired theoretical knowledge, - has an active and responsible approach to completing tasks, - shows independence in solving more complex problems.	

Brief syllabus:

1. Introduction to robotic systems, platform stability solutions, CLAWAR machines.
2. Mobile robots - wheeled, tracked and biologically inspired systems.
3. Stationary robots - manipulation systems, relative positioning, transformations.
4. Visual positioning systems - properties, principle of operation, use.
5. Possibilities of precise positioning of robotic systems.
6. Use of neural networks and interpolation systems in positioning.
7. General principles of navigation of mobile robots, processing of navigation variables of external sensors.
8. Autonomous robotic systems and their interactive interaction with the environment.
9. Sensory system of autonomous robots, multisensory approach.
10. Computer vision – laser, camera, infrared and ultrasound principles, based mainly on trigonometric principles.
11. Basic problems of mobile robotics – navigation in known and unknown environments.
12. Planning the route of the mobile robot, creating maps, avoiding obstacles.
13. Cooperation of mobile robots using centralized and distributed control.

Literature:

1. STUART, R. - NORVIG, P.: Mesterséges intelligencia modern megközelítésben Budapest :
2. Panem Könyvkiadó, 2005. 1206 s. ISBN 963 545 411 2.
3. KULCSÁR, B.: Robottechnika LSI Oktatóközpont, 2003. 394 s. ISBN 963 577 243 2.
4. CSEREY, G. – ISTENES, Z.: Autonom Mobil Robotok. Budapest: Eötvös Loránd Tudományegyetem, 2019. ISBN 978-963-284-467-1. http://dtk.tankonyvtar.hu/xmlui/handle/123456789/3722
5. MESTER, G.: Robotika. Szeged. Szegedi Tudományegyetem, 2011. ISBN 978-963-279-515-7. http://dtk.tankonyvtar.hu/xmlui/handle/123456789/7525
6. PIGLERNÉ, L. R. – STARKNÉ, W. A.: Ágens-technológia. Pannon Egyetem, 2011. http://dtk.tankonyvtar.hu/xmlui/handle/123456789/7529
7. LACZIK, B.: Robottechnika. EDUTUS Főiskola, 2012. http://dtk.tankonyvtar.hu/xmlui/handle/123456789/11920
8. SZABÓ, Z. – BUDAI, C. – KOVÁCS, L. – LIPOPVSKI, G.: Robotmechanizmusok. BME, 2014. http://dtk.tankonyvtar.hu/xmlui/handle/123456789/3421

Language, knowledge of which is necessary to complete a course:

Hungarian or Slovak

Notes:

Student workload distribution:
60% - participation in tutorials, preparation for examinations,
40% - studying literature, practicing the acquired knowledge, working on programming tasks.

Evaluation of subjects

Total number of evaluated students: 0

A	B	C	D	E	FX
0.0	0.0	0.0	0.0	0.0	0.0

Teacher: Dr. habil. András Molnár, PhD., Ing. Ondrej Takáč, PhD.

Date of last update: 02.03.2022

Approved by:

INFORMATION SHEET

Name of the university: J. Selye University	
Name of the faculty: Faculty of Education	
Code: KINF/ SMP/22	**Name:** Social, moral and legal context of computer systems development
Types, range and methods of educational activities: **Form of study:** Lecture / Seminar / Practical **Recommended extent of course (in hours):** **Per week:** 0 / 0 / 2 **For the study period:** 0 / 0 / 26 **Methods of study:** present	
Number of credits: 3	
Recommended semester/trimester of study: 2.	
Level of study: I.	
Prerequisites:	
Conditions for passing the subject: During the semester, students will complete two written quizzes for which they may earn 100% of the total points. For assessment A should be obtained at least 90 points, for assessment B at least 80 points, for assessment C at least 70 points, for assessment D at least 60 points, for assessment E at least 50 points. Credits for the subject will not be awarded to a student who does not obtain at least 50 points.	
Results of education: Educational outcomes - knowledges: Upon successful completion of the course, the student: - will be aware of the social, moral, legal and economic contexts of his/her profession, - acquire a basic knowledge of the methods and means of ensuring safety on Internet security, and acquire basic knowledge in selected areas of law application, - gain knowledge of the understanding of ICT and the information revolution. Educational outcomes - skills: Upon successful completion of the course, the student: - can use selected legal norms, - can use the Internet and means of communication in accordance with the protection of personal data, with the protection of business privacy and general security, - can use electronic signature. Educational outcomes - competencies: Upon successful completion of the course, the student will be able to make independent and correct decisions and act in specific ethical and cybercrime situations.	
Brief syllabus: 1. Legal protection of computer software 2. Copyright protection of computer programs 3. Social context of informatics and information and communication technologies 4. Information and communications technology law 5. Software piracy 6. Cybercrime 7. Legal regulations for e-commerce in Slovakia 8. Electronic signature 9. Internet safety 10. Legal issues and the internet in Slovakia	

11. Domain name registration
12. General Data Protection Regulation
13. Cross-border flow of personal data

Literature:

1. CRUME, J.: Az internetes biztonság belülről : Amit a hekkerek titkolnak. Addison-Wesley, 2003. 302 s. ISBN 9639131512.
2. CHLIPALA, M. a kol.: Právo informačných a komunikačných technológií. Slovenská technická univerzita, 2005. 186 s. ISBN 9788022721950.
3. HANCE, O. - SISÁK, G.: Üzlet és jog az interneten. Budapest : Panem, 1997. 410 s. ISBN 963545127x.
4. KYAS, O. - INOTAI, L.: Számítógépes hálózatok biztonságtechnikája. Budapest : Kossuth, 2000. 312 s. ISBN 9630941538.
5. MAISNER, M. a kol.: Základy práva informačných technológií. IURA EDITION, 2013. 320 s. ISBN 9788080785949.
6. MAISNER, M.: Základy softwarového práva. Wolters Kluwer, 2011. 356 s. ISBN 978-80-7357-638-7.
7. POLČÁK, R.: Internet a proměny práva, Vydavateľstvo: Auditorium, 2012. 388 s. ISBN 978-80-87284-22-3.
8. POLČÁK, M. Právo na internetu, Spam a odpovědnost ISP. Computer Press, 2007. 150 s. ISBN 8025117774.
9. RHEE, Y. M.: Internet Security. Wiley, 2003. 408 s. ISBN 0470852852.
10. Andrew M. St. Laurent (2004) Understanding Open Source and Free Software Licensing. O'Reilly Media, Inc. ISBN: 9780596005818.

Language, knowledge of which is necessary to complete a course:
Hungarian or Slovak

Notes:
Student workload distribution:
60% - participation in tutorials, preparation for examinations,
40% - studying literature, practicing the acquired knowledge.

Evaluation of subjects
Total number of evaluated students: 0

A	B	C	D	E	FX
0.0	0.0	0.0	0.0	0.0	0.0

Teacher: László Marák, PhD., doc. RNDr. József Bukor, PhD.

Date of last update: 02.03.2022

Approved by:

INFORMATION SHEET

Name of the university: J. Selye University	
Name of the faculty: Faculty of Education	
Code: KINF/ TAP/22	**Name:** Educational software development
Types, range and methods of educational activities: **Form of study:** Lecture / Seminar / Practical **Recommended extent of course (in hours):** **Per week:** 0 / 0 / 2 **For the study period:** 0 / 0 / 26 **Methods of study:** present	
Number of credits: 3	
Recommended semester/trimester of study: 4.	
Level of study: I.	
Prerequisites:	
Conditions for passing the subject: During the semester, students create their own applications (projects), under the guidance of the teacher. They are continuously checked and scored in their creative work as they progress with their project. The subject ends with an exam. Students must obtain at least 50% of the interim assessment (creation of own project) to be allowed to take the exam. Students are classified according to the obtained average from the overall evaluation of the interim assessment (work on the project) during the semester (50%) and the assessment of the final project (50%), which they must present in the exam. To receive grade A in the course, student must obtain at least 90%, for grade B at least 80%, for grade C at least 70%, for grade D at least 60% and for grade E at least 50%.	
Results of education: Knowledge: After completing the subject, the student: • knows strategies, methods and forms of creating applications; • knows the principles of program creation in programming languages; • knows and knows how to effectively apply the acquired programming knowledge during the development of own pedagogical applications; • knows the basic principles of creating algorithms and program structures. Skills: After completing the subject, the student: • is able to analyze and solve problems using a programming language; • is able to independently apply an algorithm to solve a given problem; • is able to controls programming actively in a given programming environment, or can integrate more environments to achieve the set goal; • is able to design and implement own project; • is capable of independent creation of presentations of teaching material, creation of e-learning courses in various environments; • is able to create applications for interactive whiteboards useable in the pedagogical process. Competencies: After completing the subject, the student: • has a high degree of independence in creation of applications (projects);	

• has an overview of the possibilities of individual tools and environments for application development;
• knows how to work effectively and independently;
• is characterized by creative thinking and independence;
• applies a creative computational thinking in his or her work;
• has an overview of the possibilities of programming and development environments for creation own applications;
• has an active programming experience;
• has an active and responsible approach to completing tasks within the subject.

Brief syllabus:

1. Possibilities of the computer as a didactic tool in individual forms and phases of teaching.
2. Presenting the learning material in different environments, choosing the topic of own application.
3. Design and implementation of own project (application).
4. Pedagogical transformation and clarity.
5. Ensuring dynamism and interactivity.
6. Creation of feedbacks in didactic applications.
7. Creation of database test systems.
8. Test creation algorithms, selection criteria.
9. Possibilities of creating an animation in different environments.
10. Possibilities of creating an interactive user interface in different environments.
11. Multimedia possibilities of individual tools and environments.
12. Possibilities of the interactive whiteboard to apply the use of the application.
13. Testing and Debugging.

Literature:

1. CZAKÓOVÁ, K. – STOFFOVÁ, V. Kreativitás és az aktív tanulást támogató programkörnyezetek. In: Mikrovilág alkalmazások : Egyetemi tankönyv. 1. kiadás. Komárno : Univerzita J. Selyeho, 2016. s. 12-31. ISBN 978-80-8122-191-0.
2. CZAKÓOVÁ, K. Saját alkalmazás fejlesztése Imagine programkörnyezetben. In: Mikrovilág alkalmazások : Egyetemi tankönyv. 1. kiadás. Komárno : Univerzita J. Selyeho, 2016. s. 35-107. ISBN 978-80-8122-191-0.
3. STOFFOVÁ, V. – CZAKÓOVÁ, K.: Prostredie na učenie sa bádaním. In: Úvod do programovania v prostredí mikrosvetov : Vysokoškolská učebnica. Komárno : Univerzita J. Selyeho, 2016. 115 s. ISBN 978-80-8122-170-5.
4. (ACD) STOFFOVÁ, V. – CZAKÓOVÁ, K.: Tvorba vlastných aplikácií v Imagine. In: Úvod do programovania v prostredí mikrosvetov : Vysokoškolská učebnica. Komárno : Univerzita J. Selyeho, 2016. 115 s. ISBN 978-80-8122-170-5.
5. STOFFOVÁ, VERONIKA: POČÍTAČ – UNIVERZÁLNY DIDAKTICKÝ PROSTRIEDOK. 1. vyd., Fakulta prírodných vied UKF, Nitra, 2004. ISBN 80-8050-765-1.
6. MOODLE: MOODLE DOCS 2.8 [online]. 2014. Dostupné na adrese: <https://docs.moodle.org/28/en/Main_page>.
Odborné články:
CZAKÓOVÁ, K.: Developing algorithmic thinking by educational computer games. In. Proceedings of the 16th International Scientific Conference: “eLearning and Software for Education : eLearning sustainment for never-ending learning. Volume 1, DOI: 10.12753/2066-026X-20-003, 2020/1, p. 26-33. Bucharest : “CAROL I” National Defence University Editura, Universitara, 2020. ISSN 2066-026X, ISSN-L 2066-026X, ISSN CD 2343 – 7669. (Scopus)

CZAKÓOVÁ, K. Mathematical Model Based Interactive Simulations In Education. In. ICERI 2019 Proceedings of the 12th International Conference of Education, Research and Innovation : Enlightening Minds through Education. DOI: 10.21125/iceri.2019.2479, p. 10120-10125, Seville : IATED Academy, 2019. ISBN 978-84-09-14755-7. ISSN 2340-1095. (WOS)
CZAKÓOVÁ, K.: Interaktív modellek és szimulációk az oktatásban. In. XXXII. Didmattech 2019 - Proceedings – New Methods and Technologies in Education and Practice : III New Methods and Tools in Education. Trnava : Trnavská univerzita v Trnave, 2019. ISBN (on line) 978 80 568 0398 1.
CZAKÓOVÁ, K.: Microworld environment of small language as „living laboratory" for developing educational games and applications. In. Proceedings of the 13th International Scientific Conference „eLearning and Software for Education" : Could technology support learning efficiency? Volume 1, DOI: 10.12753/2066-026X-17-042, 2017/1, p. 286-291. Bucharest : "CAROL I" National Defence University Publishing House, 2017. ISSN 2066-026X ISSN-L, 2066-026X, ISSN CD 2343 – 7669.

Language, knowledge of which is necessary to complete a course:
Hungarian or Slovak

Notes:
Distribution of the student's workload:
40% of the workload - direct teaching, preparation for the exam.
60% of the workload - studying the literature, work on the semester project.

Evaluation of subjects
Total number of evaluated students: 0

A	B	C	D	E	FX
0.0	0.0	0.0	0.0	0.0	0.0

Teacher: Dr. habil. Sándor Szénási, PhD., RNDr. Štefan Gubo, PhD.

Date of last update: 02.03.2022

Approved by:

INFORMATION SHEET

Name of the university: J. Selye University

Name of the faculty: Faculty of Education

Code: KINF/TFJ/22 | **Name:** Formal languages and automata

Types, range and methods of educational activities:
Form of study: Lecture / Seminar / Practical
Recommended extent of course (in hours):
Per week: 1 / 2 / 0 **For the study period:** 13 / 26 / 0
Methods of study: present

Number of credits: 4

Recommended semester/trimester of study: 4.

Level of study: I.

Prerequisites:

Conditions for passing the subject:
Interim assessment during the semester: 50% of the total assessment. During the semester two written tests with maximum score of 15 points per test will be held. During the semester, students independently work on 2 practical assignments (1 from the part of regular languages and finite automata, and 1 from the part of context-free languages and pushdown automata), for their submission a total of 20 points can be obtained. Students must obtain at least 50% of the interim assessment to be allowed to take the exam.
Exam: 50% of the total assessment. The course is finished by written exam, on which 50 points can be obtained. To successfully pass the exam, it is necessary to obtain at least 50% of the exam evaluation.
The overall assessment consists of the sum of points from the interim assessment and the final exam. To receive grade A in the course, student must obtain at least 90 points, for grade B at least 80 points, for grade C at least 70 points, for grade D at least 60 points and for grade E at least 50 points. Credits will not be given to a student who obtain less than 50 points.

Results of education:
Knowledge:
After completing the subject, the student:
• has theoretical knowledge of the theory of formal languages and automata.
Skills:
After completing the subject, the student:
• is able to construct regular grammars, finite automata and regular expressions,
• is able to construct context-free grammars and pushdown automata,
• is able to prove about given languages that they are not regular or context free,
• knows the rules of creation of documentations for practical tasks.
Competencies:
After completing the subject, the student:
• is able to work independently and efficiently,
• has an active and responsible approach to completing tasks within the subject.

Brief syllabus:
1. Introduction to the Theory of Formal Languages and Automata, basic terms and definitions.
2. Chomsky hierarchy of languages, Chomsky hierarchy of grammars.

3. Regular languages – basic terms.
4. Nondeterministic and deterministic finite automata.
5. Connection between nondeterministic and deterministic finite automata
6. Connection between regular grammars and finite automata.
7. Regular expressions.
8. Pumping lemma for regular languages.
9. Context-free languages – basic terms.
10. Pushdown automata.
11. Connection between context-free grammars and pushdown automata.
12. Pumping lemma for context-free languages.
13. Top-down parsing, bottom-up parsing.

Literature:

1. GUBO, Š.: Formális nyelvek és automaták. Komárno : Univerzita J. Selyeho, 2015, 131 s. ISBN 978-80-8122-148-4.
2. FÜLÖP, Z.: Formális nyelvek és szintaktikus elemzésük. Szeged : Polygon, 1999, 124 s. ISSN 1417-0590.
3. BACH, I.: Formális nyelvek. Budapest : Typotex, 2005, 227 s. ISBN 978-963-9132-92-4.
4. ROVAN, B. - FORIŠEK, M.: Formálne jazyky a automaty. Bratislava : Univerzita Komenského, 2013. 125 s. Dostupné na: http://foja.dcs.fmph.uniba.sk/materialy/skripta.pdf
5. SINGH, A.: Elements of Computation Theory. London : Springer-Verlag, 2009. 422 s. ISBN 978-1-84882-496-6.
6. HOPCROFT, J. E. – MOTWANI, R. – ULLMAN, J. D.: Introduction to Automata Theory : Languages, and Computation. London : Pearson, 2014. 488 s. ISBN 978-1-292-03905-3.
7. HORVÁTH, G. – NAGY, B.: Formal Languages and Automata Theory. Budapest : Typotex, 2014, 135 s. ISBN 978-963-2793-44-3.

Language, knowledge of which is necessary to complete a course:
Hungarian or Slovak

Notes:
Distribution of the student's workload:
50% of the workload - direct teaching, preparation for the tests and the exam.
50% of the workload - studying the literature, practicing the acquired knowledge, work on practical assignments.

Evaluation of subjects
Total number of evaluated students: 0

A	B	C	D	E	FX
0.0	0.0	0.0	0.0	0.0	0.0

Teacher: prof. RNDr. Tibor Kmeť, CSc., RNDr. Štefan Gubo, PhD.

Date of last update: 02.03.2022

Approved by:

INFORMATION SHEET

Name of the university: J. Selye University	
Name of the faculty: Faculty of Education	
Code: KINF/ TMA/22	**Name:** Multimedia application development
Types, range and methods of educational activities: **Form of study:** Lecture / Seminar / Practical **Recommended extent of course (in hours):** **Per week:** 0 / 0 / 2 **For the study period:** 0 / 0 / 26 **Methods of study:** present	
Number of credits: 2	
Recommended semester/trimester of study: 6.	
Level of study: I.	
Prerequisites:	
Conditions for passing the subject: During the semester, students solve practical problems for which they can get 60 points. At the end of the semester, students will complete a term project for which they may receive 40 points. A grade of A requires at least 90% points, a grade of B requires at least 80% points, a grade of C requires at least 70% points, a grade of D requires at least 60% points, and a grade of E requires at least 50% points. Credit will not be awarded to a student who has not earned 50% of the points at the end of the semester.	
Results of education: Knowledge: After completing the course, the student will be familiar with the SVG format. The student knows how to create and modify an SVG image using program code or a vector graphics editor. The student knows the possibilities of animating, interacting, and attaching sound to vector images. Skills: Upon completion of this course, students will have basic knowledge and skills in vector image (graphical object) creation and simple animations in an SVG/JavaScript or SVG/ JQuery environment. Upon completion of the course, students are able to independently create illustrations and simple animations that can be easily integrated into web pages using HTML. Competencies: Upon completion of the course, students have the ability to create vector-based audiovisual interactive web applications. The student can use his/her skills as a web developer, as a full-stack developer, as a web game developer, or as a web animator. The student can also use his/her skills as a developer of a system for visualizing information, for creating graphs, charts, infographics, and for visually representing information.	
Brief syllabus: 1. Basics of vector representation. SVG format. Basic objects supported in SVG format, lines, segments, circles, ellipses, rectangles, spline-y. 2. Object attributes such as color, transparency, frame, position, orientation. 3. Grouping of objects. 4. Software for creating SVG images, integration and display of SVG images in web pages. 5. Creating simple SVG images.	

6. Vectorization (framing of images).
7. Basic image animation, panning and rotation.
8. Advanced animation using external libraries, acceleration, deceleration, vibration, jumping.
9. Morphing images.
10. Creating charts.
11. Interactivity in SVG.
12. Complex interaction with an image.
13. Connecting sound to animation.

Literature:

1. EISENBERG, J. D.: Amelia Bellamy-Royds (2014), SVG Essentials, 2nd Edition. O'Reilly Media, Inc. ISBN: 9781449374358
2. MACRAE, C (2013). Learning from jQuery. O'Reilly Media, Inc. ISBN: 9781449335199
3. BAH, T.: (2011). Inkscape: Guide to a Vector Drawing Program, 4th Edition. Pearson.

Language, knowledge of which is necessary to complete a course:
Hungarian or Slovak

Notes:
Student workload distribution:
55% - participation in classes, preparation for exercises,
45% - studying literature, practicing the acquired knowledge, working on practical assignments, preparing the term paper.

Evaluation of subjects
Total number of evaluated students: 0

A	B	C	D	E	FX
0.0	0.0	0.0	0.0	0.0	0.0

Teacher: Dr. habil. Attila Elemér Kiss, CSc., László Marák, PhD.

Date of last update: 02.03.2022

Approved by:

INFORMATION SHEET

Name of the university: J. Selye University

Name of the faculty: Faculty of Education

Code: KINF/TPS/22 | **Name:** Pedagogical software creation

Types, range and methods of educational activities:
Form of study: Lecture / Seminar / Practical
Recommended extent of course (in hours):
Per week: 0 / 0 / 2 **For the study period:** 0 / 0 / 26
Methods of study: present

Number of credits: 2

Recommended semester/trimester of study: 6.

Level of study: I.

Prerequisites:

Conditions for passing the subject:
During the semester students independently solve a designated programming task - semester project, the output of which is their own pedagogical software. Students also have the opportunity to consult their project during the semester (its specific development phase) with the teacher. The students' work on the project is continuously monitored and scored. The completed pedagogical software will be handed over at the end of the semester (electronically and with instructions for use).
The subject ends with exam. The condition for admission to the exam is that the score obtained from the continuous evaluation of the preparation of the student's own project (pedagogical software) reaches at least 50% of the possible points. Students are evaluated on the basis of the average obtained from the evaluation of the work (project work) during the semester (50%) and the level of the final work (project) (50%). They must defend the project during the exam.
An average of at least 90% is required to achieve grade A, at least 80% for grade B, at least 70% for grade C, at least 60% for grade D, and at least 50% for grade E.

Results of education:
Knowledge:
After completing the subject, the student:
• knows the strategies, methods and forms of pedagogical software development;
• is aware of the possibilities of the computer in the individual teaching forms and phases;
• knows the basic principles of program creation in programming languages;
• knows the basic principles of creating algorithms and program structures for controlling;
• knows and can effectively apply the acquired programming knowledge to create own pedagogical software;
• knows the pedagogical and psychological aspects of pedagogical software creation.
Skills:
After completing the subject, the student:
• can analyze problems and solve them using a programming language;
• can choose the right algorithm to solve the problem;
• can actively write program in a specific programming environment or can integrate several environments to achieve the set goal;
• is able to select and skillfully use the appropriate programming environment (or even combine different environments) for the creation of pedagogical software on a given topic;

• is able to plan and implement his own project;
• knows the basics of pedagogical software creation;
• is aware of the rules for preparing the correct documentation of software products;
• is able to independently prepare the presentations of the course material;
• can use and apply the created software in the pedagogical process.
Competencies:
After completing the subject, the student:
• demonstrates a high degree of independence in creating the software (project);
• has an overview of the possibilities of individual tools and environments required for the creation of pedagogical software;
• knows how to work effectively independently;
• is characterized by creative thinking and independence;
• applies a creative IT way of thinking in his work;
• has an overview of the possibilities of programming and development environments for creating own software;
• is characterized by active programming experience;
• has an active and responsible attitude towards the completion of subject tasks.

Brief syllabus:

1. Possibilities of using the computer in certain forms and phases of teaching.
2. Presentation of learning material - knowledge related to computer use.
3. Pedagogical and psychological aspects of pedagogical software creation.
4. Classification of pedagogical software according to different aspects.
5. Selection of the topic of the pedagogical software, project planning.
6. Implementation of pedagogical software.
7. Computer knowledge testing.
8. Possibilities for making animations in different programming languages.
9. Possibilities of creating an interactive user interface.
10. Possibilities of multimedia in the learning process.
11. Software testing and fine-tuning.
12. Rules for creating documentation for software products.
13. Testing of final products in real conditions.

Literature:

1. ANGSTER, E.: Az objektumorientált tervezés és programozás alapjai. Budapest : Akadémiai, 2000. 312 s.ISBN 9636508186.
2. CHAPMAN, N. - CHAPMAN, J.: Digitel multimedia: Second Edition, 2003. 700 s. ISBN 0470858907.
3. CZAKÓOVÁ, K. – STOFFOVÁ, V. Kreativitás és az aktív tanulást támogató programkörnyezetek. In: Mikrovilág alkalmazások : Egyetemi tankönyv. 1. kiadás. Komárno : Univerzita J. Selyeho, 2016. s. 12-31. ISBN 978-80-8122-191-0.
4. CZAKÓOVÁ, K. Saját alkalmazás fejlesztése Imagine programkörnyezetben. In: Mikrovilág alkalmazások : Egyetemi tankönyv. 1. kiadás. Komárno : Univerzita J. Selyeho, 2016. s. 35-107. ISBN 978-80-8122-191-0.
5. KADLEC, V.: Učíme se programovat v Delphi a jazyce OBJECT PASCAL. Brno : Computer Press, 2002. 290 s. ISBN 8072262459.
6. MCCARTHY, J.: Softwarové projekty. 1. vyd. Praha : Computer Press, 1999. 190 s. ISBN 80-7226-164-0.
7. MCCARTHY, J.: Softwarové projekty. Brno : Computer Press, 1999. 190 s. ISBN 8072261940.

8. STOECKER, M.: Developing Wisndows-Based Applicatoins with Microsft. Microsoft .net, 2003. 600 s. ISBN 0735619263.
9. STOFFOVÁ, V. – CZAKÓOVÁ, K.: Prostredie na učenie sa bádaním. In: Úvod do programovania v prostredí mikrosvetov : Vysokoškolská učebnica. Komárno : Univerzita J. Selyeho, 2016. 115 s. ISBN 978-80-8122-170-5.
10. STOFFOVÁ, V. – CZAKÓOVÁ, K.: Tvorba vlastných aplikácií v Imagine. In: Úvod do programovania v prostredí mikrosvetov : Vysokoškolská učebnica. Komárno : Univerzita J. Selyeho, 2016. 115 s. ISBN 978-80-8122-170-5.
11. SZIRMAY-KALOS, L. - LÁSZLÓ, Z. – KONDOROSI, K.: Objektum-orientált szoftverfejlesztés. Budapest : ComputerBooks, 2001. 427 s. ISBN 963 618 108 X.
12. SZIRMAY-KALOS, L. Háromdimenziós grafika, animáció és játékfejlesztés. Budapest : ComputerBooks, 2004. 486 s. ISBN 9636183031.
13. VÁMOSSY, Z.: Delphi a gyakorlatban. Bicske : Szak, 2002. 132 s. ISBN 963 9131 22 9.
14. VÉG, Cs.: Alkalmazásfejlesztés : a Unified Modeling Language szabványos jelöléseivel. Debrecen : Logos 2000, 1999. 246 s. ISBN 963 03 7660 1.
Odborné články:
CZAKÓOVÁ, K.: Developing algorithmic thinking by educational computer games. In. Proceedings of the 16th International Scientific Conference: “eLearning and Software for Education : eLearning sustainment for never-ending learning. Volume 1, DOI: 10.12753/2066-026X-20-003, 2020/1, p. 26-33. Bucharest : “CAROL I” National Defence University Editura, Universitara, 2020. ISSN 2066-026X, ISSN-L 2066-026X, ISSN CD 2343 – 7669. (Scopus)
CZAKÓOVÁ, K. Mathematical Model Based Interactive Simulations In Education. In. ICERI 2019 Proceedings of the 12th International Conference of Education, Research and Innovation : Enlightening Minds through Education. DOI: 10.21125/iceri.2019.2479, p. 10120-10125, Seville : IATED Academy, 2019. ISBN 978-84-09-14755-7. ISSN 2340-1095. (WOS)
CZAKÓOVÁ, K.: Interaktív modellek és szimulációk az oktatásban. In. XXXII. Didmattech 2019 - Proceedings – New Methods and Technologies in Education and Practice : III New Methods and Tools in Education. Trnava : Trnavská univerzita v Trnave, 2019. ISBN (on line) 978 80 568 0398 1.
CZAKÓOVÁ, K.: Microworld environment of small language as „living laboratory” for developing educational games and applications. In. Proceedings of the 13th International Scientific Conference „eLearning and Software for Education“ : Could technology support learning efficiency? Volume 1, DOI: 10.12753/2066-026X-17-042, 2017/1, p. 286-291. Bucharest : “CAROL I” National Defence University Publishing House, 2017. ISSN 2066-026X ISSN-L, 2066-026X, ISSN CD 2343 – 7669.

Language, knowledge of which is necessary to complete a course:
Hungarian or Slovak

Notes:
Student workload:
55% - participation in lessons, preparation for the exam,
45% - study of literature, completion of programming tasks, preparations for semester work.

Evaluation of subjects
Total number of evaluated students: 0

A	B	C	D	E	FX
0.0	0.0	0.0	0.0	0.0	0.0

Teacher: Dr. habil. Dr. Gábor Kiss, PhD., PaedDr. Krisztina Czakóová, PhD.
Date of last update: 02.03.2022
Approved by:

INFORMATION SHEET

Name of the university: J. Selye University	
Name of the faculty: Faculty of Education	
Code: KINF/ TXE/22	**Name:** Text editors
Types, range and methods of educational activities: **Form of study:** Lecture / Seminar / Practical **Recommended extent of course (in hours):** **Per week:** 0 / 0 / 2 **For the study period:** 0 / 0 / 26 **Methods of study:** present	
Number of credits: 3	
Recommended semester/trimester of study: 1.	
Level of study: I.	
Prerequisites:	
Conditions for passing the subject: Students are required to actively participate in class, which is monitored and counts for 20% of the overall course grade. During the semester, students work on their two semestral works, which are required and must be turned in for grading. The course ends with an exam. Grading is determined by the average of the 2 graded semestral works, each of which students must complete at least 50%. The student is classified according to the average obtained in the tests and its defense (80% of the total grade) and active participation in the exercises (20% of the total grade). To obtain grade „A“ students have to obtain minimum 90% of the total score, to obtain grade „B“ students have to obtain 80% of the total score, to obtain grade „C“ students have to obtain 70% of the total score, to obtain grade „D“ students have to obtain 60% of the total score, to obtain grade „E“ students have to obtain 50% of the total score. There is no credit for the subject if a student obtains less than 50%.	
Results of education: Knowledge: The student has knowledge of word processors, proficiency in the LaTex word processor. Skills: The student is able to use the LaTex word processor, is proficient in word processing, and is able to program in LaTex at a basic level. The student is proficient in the basic principles of creating structured text documents in the TeX typographic system (LaTeX). Competences: The student is characterized by independence in working in Latex at the user level.	
Brief syllabus: 1. The environment of text editors and their basic functions. 2. Text editors (types of text editors: interpreters). 3. The environment of text editors and their basic functions. 4. Document creation standards. Basic document structure. 5. Introduction to TeX. 6. Writing plain texts, choice of font size and type. 7. LaTeX environments for creating lists, bullets, tables, simple charts.	

8. Setting mathematical formulae, equations, matrices, etc. Cross-referencing.
9. Fundamentals of TeX programming.
10. Creating simple macros.
11. Incorporating graphics into text.
12. Creating presentations.
13. Elaboration of the selected topic and independent creation of a professional text.

Literature:

1. STOFFA,V. - CSÍZI,L. - SZŐKÖL, I. - TÓTH, K. - VÉGH,L.: Az informatika alapjai I. Komárno: UJS, 2007, s. 268. ISBN 978-80-89234-29-5.
2. STOFFOVÁ, V. - CSÍZI, L. - TÓTH, K. - SZŐKÖL, Š.: Informačné a komunikačné technológie v praxi II. Komárno : Univerzita J. Selyeho, 2007, s. 316. ISBN 978-80-89234-42-4.
3. STOFFOVÁ, V. - CSÍZI, L. - TÓTH, K. - SZŐKÖl, Š.: Információs és kommunikációs technológiák a gyakorlatban II. Komárno : Univerzita J. Selyeho, 2007, s. 316. ISBN 978-80-89234-69-1.
4. RYBIČKA, J.: LATEX pro začátečníky. 3. vyd. Brno : nakladatelství KONVOJ, spol. s.r.o., 2003. 239 s. ISBN 80-7302-049-1.
5. WETTL, F. – MAYER, Gy.: Latex kézikönyv : Könnyen is lehet! 1. vyd. Budapest : Panem, 2004. 768 s. ISBN 963 545 398 1.

Language, knowledge of which is necessary to complete a course:
Hungarian or Slovak

Notes:
Student workload distribution:
45% - attendance at tutorials, exam preparation,
55% - studying literature, practicing acquired knowledge, working on practical assignments, preparing term papers.

Evaluation of subjects
Total number of evaluated students: 0

A	B	C	D	E	FX
0.0	0.0	0.0	0.0	0.0	0.0

Teacher: RNDr. József Udvaros, PhD., PaedDr. Márk Csóka

Date of last update: 02.03.2022

Approved by:

INFORMATION SHEET

Name of the university: J. Selye University	
Name of the faculty: Faculty of Education	
Code: KINF/ UDI/22	**Name:** Introduction to IT
Types, range and methods of educational activities: **Form of study:** Lecture / Seminar / Practical **Recommended extent of course (in hours):** **Per week:** 2 / 1 / 0 **For the study period:** 26 / 13 / 0 **Methods of study:** present	
Number of credits: 4	
Recommended semester/trimester of study: 1.	
Level of study: I.	
Prerequisites:	
Conditions for passing the subject: During the semester, students write two written papers, which are evaluated as a percentage. Students must achieve a score of at least 50% for both written papers in order to take the exam. During the semester, students work independently on a semester assignment or project (processing of literature), and they can receive a total of 20 points out of a total of 100 points for their submission and presentation. The combined exam consists of a written and oral part. To pass the exam, students must achieve at least 50% in the oral exam. The students are classified based on the obtained average, which includes the continuous performance of the semester, the work of the semester project and the result of the exam. A grade is at least 90 points, B grade is at least 80 points, C grade is at least 70 points, D grade is at least 60 points and E grade is at least 50 points. A student who scores less than 50 points cannot receive credit.	
Results of education: Educational results - knowledge: After completing the subject, the student: • knows the role, tools and solutions of IT; • knows the concept of data and the principles of information and processing; • knows the basic principles of coding and displaying information on a computer and simple methods for their processing; • knows the principles of basic mathematical operations and conversions performed in the binary system, as well as the floating-point representation of real numbers; • knows the connections between individual number systems (conversions, transformations); • can think algorithmically and apply knowledge to solve the problem; • knows the basic principles of creating and graphically representing formal algorithms; • knows the classification of programming languages, the structure of the given programming language (e.g. Python), knows the elements and programming structures of the given language (condition, cycle, functions). Learning outcomes - skills: After completing the subject, the student: • can analyze and solve problems using a programming language; • able to independently apply an algorithm to solve a specific problem;	

• can divide the problem into smaller sub-problems;
• can think algorithmically;
• capable of conversions between number systems;
• able to recognize the repetitive tasks of the problem and solve them with the necessary program structure (cycle, state);
• able to select the necessary data structures in the program;
• able to design and implement a program in the given programming language.
Educational results - competences:
After completing the subject, the student:
• shows a high degree of independence in solving problems and creating programs to solve them;
• characterized by creative thinking and independence;
• uses creative IT and algorithmic thinking to solve problems;
• can explain everyday IT problems;
• has an active and responsible attitude towards the completion of subject tasks

Brief syllabus:
1. Definition of basic concepts, IT tools, storage, data processing, algorithm.
2. Graphic representation of the algorithm, with a branching condition.
3. Graphic representation of the algorithm using cycles.
4. Graphic representation of the algorithm, solving complex tasks.
5. Number systems, conversions.
6. Number systems, basic mathematical operations.
7. Numerical representations, fixed precision representation, floating precision representation.
8. Program creation process, planning, steps, development cycle.
9. Classification of programming languages, work in a specific programming language (e.g Python), concepts and structure.
10. Programming algorithms with branching.
11. Programming algorithms using cycles.
12. Programming algorithms using functions and procedures.
13. Programming complex algorithms, displaying functions, drawing using programming tools.

Literature:
1. ANNUS, G.: Informatikai alapok. Szeged : JGYF Kiadó, 2001. 204 s. ISBN 0991508.
2. KATONA, Gy.: A számítástudomány alapjai. Budapest : Typotex Elektronikus Kiadó Kft., 2002. 192 s. ISBN 963 9326 24 0.
3. KOVÁCS, M.: Bevezetés a Számítástechnikába. Budapest : LSI Oktatóközpont, 2002. 368 s. ISBN 963 577 270 X.
4. STOFFA, V.: Az informatika alapjai I. (Základy informatiky). Apáczai közalapítvány, 2007. 268 s. ISBN 978-80-89234-29-5.
5. STOFFA, V.: Algoritmizáció és programozás I. (Algoritmizácia a programovanie I). Komárno : Univerzita J. Selyeho v Komárne, 2005. 174 s. ISBN 80-969251-7-2.
6. STOFFOVÁ, V.: Informatika. Informačné technológie a výpočtová technika. Nitra : Prírodovedec, 2001. 230 s. ISBN 80-8050-450-4.
7. STOFFOVÁ, V.: Počítač univerzálny didaktický prostriedok. 1. vyd. Nitra, 2004. 173 s. ISBN 80 8050 765 1.

Language, knowledge of which is necessary to complete a course:
Hungarian or Slovak

Notes:
Distribution of student workload:

60% - participation in lessons, preparation for background checks and exams,
40% - study of professional literature, practice of acquired knowledge, work on practicals assignments.

Evaluation of subjects
Total number of evaluated students: 0

A	B	C	D	E	FX
0.0	0.0	0.0	0.0	0.0	0.0

Teacher: Dr. habil. Attila Elemér Kiss, CSc., PaedDr. Krisztina Czakóová, PhD.

Date of last update: 02.03.2022

Approved by:

INFORMATION SHEET

Name of the university: J. Selye University	
Name of the faculty: Faculty of Education	
Code: KINF/ ZLD/22	**Name:** Basics of air transport
Types, range and methods of educational activities: **Form of study:** Lecture / Seminar / Practical **Recommended extent of course (in hours):** **Per week:** 0 / 2 / 0 **For the study period:** 0 / 26 / 0 **Methods of study:** present	
Number of credits: 2	
Recommended semester/trimester of study: 5.	
Level of study: I.	
Prerequisites:	
Conditions for passing the subject: The evaluation of the subject consists of a theoretical and a practical part. At the end of the semester, there will be a written test with maximum score of 75 points. Within the practical part (flight on the simulator), it is possible to get a total of 25 points. To receive grade A in the course, student must obtain at least 90 points, for grade B at least 80 points, for grade C at least 70 points, for grade D at least 60 points and for grade E at least 50 points. Credits will not be given to a student who obtain less than 50 points.	
Results of education: Knowledge: After completing the subject, the student: • has theoretical knowledge about air transport and unmanned aircraft systems, the history of aviation, aerodynamics, aviation meteorology, communication and aviation law, • has the knowledge necessary to pass the theoretical part of the remote pilot examination. Skills: After completing the subject, the student: • is able to perform a flight on the DJI Phantom 3 Flight Simulator. Competencies: After completing the subject, the student: • has an active and responsible approach to completing tasks within the subject.	
Brief syllabus: 1. Introduction to the subject Basics of Aviation, history of aviation. 2. Aircraft general knowledge. 3. The basics of flight – aerodynamics. 4. Airplane performance and flight planning. 5. Aviation meteorology. 6. A METAR report and TAF forecast. 7. Communication. 8. Aerodromes, interesting facts about airports. 9. Airspace, the ICAO map. 10. Operational procedures.	

11. Aviation law and air traffic control procedures.
12. Unmanned aircraft systems (UAS), applications of UAS.
13. Flight on the simulator DJI Phantom 3 Flight Simulator.

Literature:

1. KELLER, L et al.: Učebnice pilota 2016. Příbram : Svět křídel, 2016. 408 s. ISBN 978-80-87567-89-0.
2. Letecká mapa ICAO Slovenska 2016.
3. FÁBIÁN, A.: PPL kézikönyv : A repülőgép-vezetés elmélete. Budapest : Skylight Cerative Ec., 2010. 466 s. ISBN 978-963-06-9062-1.
4. ATKINSON, S.: The Aircraft Book : The definitive visual history. London : Dorling Kindersley, 2013. 320 s. ISBN 978-1-4093-6480-1.
5. BEARD, R. W. – McLAIN, T. W.: Small Unmanned Aircraft : Theory and Practice. New Jersey, NJ : Princeton University Press, 2012. 300 s. ISBN 978-0-691-14921-9.
6. FEDERAL AVIATION ADMINISTRATION: Pilot’s Handbook of Aeronautical Knowledge, 2016. Dostupné na: https://www.faa.gov/regulations_policies/handbooks_manuals/aviation/phak/

Language, knowledge of which is necessary to complete a course:
Hungarian or Slovak

Notes:
Distribution of the student's workload:
60% of the workload - direct teaching, preparation for the test and the practical part (flight on the simulator).
40% of the workload - studying the literature, practicing the acquired knowledge.

Evaluation of subjects
Total number of evaluated students: 0

A	B	C	D	E	FX
0.0	0.0	0.0	0.0	0.0	0.0

Teacher: RNDr. Štefan Gubo, PhD., Ing. Ondrej Takáč, PhD.

Date of last update: 02.03.2022

Approved by:

INFORMATION SHEET

Name of the university: J. Selye University	
Name of the faculty: Faculty of Education	
Code: KINF/ŠS/22	**Name:** State exam

Types, range and methods of educational activities:
Form of study:
Recommended extent of course (in hours):
Per week: For the study period:
Methods of study: present

Number of credits: 2

Recommended semester/trimester of study: 5., 6..

Level of study: I.

Prerequisites:

Conditions for passing the subject:
All students who have met the requirements of the programme of study in the final year of their studies may take the state examination at the normal time of the study timetable.
In the oral state examination, the student will give an account of the knowledge and skills acquired in his/her field of specialisation and the interdisciplinary links with the relevant fields of specialisation. Demonstrate the ability to communicate information, ideas, problems and solutions to professional and lay audiences.
The state examination takes the form of a colloquium in which the student's performance is assessed on a scale from A to FX. The grade counts towards the overall state examination grade. The oral examination is graded on the following scale: A - 100-91%, B - 90-81%, C - 80-71%, D - 70-61%, E - 60-50%. A student who fails to achieve 50% will receive no credit.
The results of the state examination and the examination will be announced by the chairman of the board in public.

Results of education:
Knowledge:
- the student has acquired knowledge in the compulsory and profile subjects of the study programme,
- the student is able to define and interpret in his/her own words basic concepts, explain and describe basic processes, characterise and apply scientific methods of research in the areas specified in the subject's thematic plan,
- the ability to analyse and evaluate the knowledge acquired in the subject.

Skills:
- the student is able to present his/her expertise,
- the student is able to transfer his/her knowledge,
- the ability to organise and apply the theoretical knowledge acquired,
- the student has the ability to organise and apply the knowledge and to apply it in a coherent way.

Competences:
- the student is able to express his/her linguistic and professional culture in the oral examination,
- the student can use the knowledge acquired in a wider context,
- the ability to put into practice and organise the knowledge acquired,

- the student is able to use his/her knowledge in a creative way when solving problems and to analyse the problem and organise new solutions,
the student is able to answer the questions of the committee to the required standard.

Brief syllabus:
I. Fundamentals of Informatics
II. Programming

Literature:
Literature listed in the information sheets of the study programme

Language, knowledge of which is necessary to complete a course:
Hungarian or Slovak

Notes:
The state examination takes place before an examination board whose members are appointed by the dean.

Evaluation of subjects
Total number of evaluated students: 0

A	B	C	D	E	FX
0.0	0.0	0.0	0.0	0.0	0.0

Teacher:

Date of last update: 04.03.2022

Approved by: