MINISTERUL EDUCAȚIEI
UNIVERSITATEA „AUREL VLAICU“ DIN ARAD
310130 Arad, B-dul Revolutiei nr. 77, P.O. BOX 2/158 AR
Tel.: 0040-257- 283010; fax. 0040-257- 280070
http://www.uav.ro; e-mail: rectorat@uav.ro
Operator de date cu caracter personal nr. 2929

SYLLABUS

1. Study programme

1.1. Higher education institution	**„Aurel Vlaicu” University of Arad**
1.2. Faculty	**of Exact Sciences**
1.3. Department	**Department of Mathematics and Computer Science**
1.4. Field of study	**Informatics**
1.5. Study year	**2024-2025**
1.6. Study level	**Bachelor**
1.7. Study programme / Qualification	**Computer Science (in English)**
1.8. Form of education	**Full – Time study**

2. Course details

2.1. Name of the course	**GIBS5O01 Programming environments and tools**
2.2. Course coordinator	**Prof. dr. Valeriu BEIU**
2.3. Seminar/laboratory/project coordinator	**Software developer Marcela-Florina FLOREA**
2.4. Study year	**3**
2.5. Semester	**1**
2.6. Evaluation type	**ES**
2.7. Course type	**Op**

3. Estimated total time (hours per semester)

3.1. Hours per week	**4**
3.2. Lecture hours per week	**2**
3.3. Seminar/laboratory/project hours per week	**2**
3.4. Total hours per curriculum	**56**
3.5. Lecture hours per semester	**28**
3.6. Seminar/laboratory/project hours per semester	**28**
	Time division [hrs]
3.4.1. Independent study from textbooks, course support, bibliography and notes	**20**
3.4.2. Additional reading (libraries, specialized electronic platforms and field research)	**20**
3.4.3. Preparing of seminars/laboratories/projects, homework, papers, portfolios and essays	**20**
3.4.4. Tutorial coaching	**5**
3.4.5. Examinations	**4**
3.4.6. Other activities	**0**
3.7. Total individual study hours	**69**
3.8. Total hours per semester	**125**
3.9. Number of ECTS credits	**5**

4. Prerequisites (if applicable)

4.1. Curriculum related	**Advanced programming methods, OOP languages**
4.2. Competence related	**- ability to write code in a high-level programming language** **- basic database knowledge** **- basic networking concepts**

5. Conditions (if applicable)

5.1. for the lecture	- **Lecture room with: laptops, video projector, software installed**
5.2. for the seminar	
5.3. for the laboratory	- **Seminar classroom, laptops, networking, internet connection, software programs**
5.4. for the project	

6. Specific educational objectives (competences to be acquired)

6.1. Professional competencies	**C1. Programming in high level programming languages.** **C2. Development and maintenance of computer applications.** **C3. Using computer tools in interdisciplinary context.** **C4. Using the theoretical bases of computers and formal models.** **C5. Database design and database management.**
6.2. Transversal competencies	**CT1. Applying the rules of organized and efficient work, of responsible attitudes towards teaching-scientific field, to value the own creative potential, while respecting the principles and norms of professional ethics.** **CT2. Efficient conduct of the activities organized in an inter-disciplinary group and developing personal communication skills, networking and collaboration with various groups.** **CT3. Using of efficient methods and techniques for learning, informing, research and development of the capacity to value knowledge, adapting to the requirements of a dynamic society and communicating in English and in an Internationally widespread language.**

7. Course outcomes (resulting from the specific educational objectives to be acquired)

7.1. General outcomes	- **Basic knowledge on ADO.NET and ASP.NET; building applications using ADO.NET and ASP.NET** - **Understanding concepts and problems of the systems developed** - **Better knowledge of design and implement software solutions**
7.2. Specific outcomes	- **Developing the ability to design and solve problems using OOP languages, databases, tools and software environments.** - **Understanding the concepts and existing technics do develop distributed software.** - **Get used to the modern approach of developing software systems.**

8. Outline (if applicable)

8.1 Lecture Outline	Teaching methods	Remarks
1. Platforms dedicated to the development of software solutions and applications, IDE (Integrated Development Environment)	Lecture, debate, exposition, problematization, interactive exposition, web documentation, exemplification.	C1. 4 hours
2. a. Code editor; b. Code translator; c. Self documentation; d. Function libraries;. e. Building automation: tools that save time by automating processes. f. Debugger: built to detect possible errors.	Lecture, debate, exposition, problematization, interactive exposition, web documentation, exemplification.	C2. 8 hours
3. Stages of creating a software product.	Lecture, debate, exposition, problematization, interactive exposition, web documentation, exemplification.	C3. 8 hours
4. Test tools (capture/reproduce, automated test execution, coverage analyzers, test case generators, test data generators, logic/complexity analyzer, bug tracker, test management).	Lecture, debate, exposition, problematization, interactive exposition, web documentation, exemplification.	C4. 8 hours.

8.2 Lecture References

1. Benjamin Perkins, Jacob Vibe Hammer, Jon D. Reid, Beginning C# 7 Programming with Visual Studio 2017, 1st Edition, 2018;
2. Christian Nagel, Professional C# 7 and .NET Core 2.0 7th Edition, 2018;
3. Andrew Troelsen, Philip Japikse, Pro C# 7: With .NET and .NET Core 8th ed. Edition. 2018;
4. C# 4.0 - The Complete Reference, Herbert Schildt, McGraw-Hill, 2010
5. Microsoft ADO.NET 4 Step by Step, Tim Patrick,Microsoft Press; 1 edition, 2010;
6. Dorothy Graham, Rex Black, Erik van Veenendaal, Foundations of Software Testing ISTQB Certification, 4th edition, Cengage Learning EMEA; 4th edition (August 9, 2019);
7. Paul Ammann, Jeff Offutt, Introduction to Software Testing 2nd Edition, Cambridge University Press (December 13, 2016);
8. Glenford J. Myers, Corey Sandler, Tom Badgett, The Art of Software Testing 3rd Edition Wiley; 3 edition (November 8,2011);
9. Marius Tomescu - Note de curs si laborator, platforma SUMS

8.3 Seminar Outline	Teaching methods	Remarks

8.4 Seminar References

8.5 Laboratory Outline	Teaching methods	Remarks
1. Advanced notions of the C# language 4.0 Applications: Generic classes and functions Generic types Threads of execution.	The exercise, the discussions and the debate, the modeling, the project. The use of the	L1. 6 hours

	applications specialized software, web documentation, organized group work	
2. Visual studio The .NET platform	The exercise, the discussions and the debate, the modeling, the project. The use of specialized software applications, the documentation on the web, the work in an organized group.	L2. 6 hours
3 Creating software products of medium complexity.	The exercise, the discussions and the debate, the modeling, the project. The use of specialized software applications, the documentation on the web, the work in an organized group.	L3. 8 hours
4. Site security testing; website functionality test; websites, online stores, databases and web applications testing services; tools for web application performance testing; web security tests; user interface testing of web applications; windows based web testing tools;	The exercise, the discussions and the debate, the modeling, the project. The use of specialized software applications, the documentation on the web, the work in an organized group.	L4. 8 hours

8.6 Laboratory Bibliography

1. Benjamin Perkins, Jacob Vibe Hammer, Jon D. Reid, Beginning C# 7 Programming with Visual Studio 2017 1st Edition, 2018;
2. Christian Nagel, Professional C# 7 and .NET Core 2.0 7th Edition, 2018;
3. Andrew Troelsen, Philip Japikse, Pro C# 7: With .NET and .NET Core 8th ed. Edition. 2018;
4. C# 4.0 - The Complete Reference, Herbert Schildt, McGraw-Hill, 2010
5. Microsoft ADO. NET 4 Step by Step, Tim Patrick, Microsoft Press; 1 edition, 2010;
5. Dorothy Graham, Rex Black, Erik van Veenendaal, Foundations of Software Testing ISTQB Certification, 4th edition, Cengage Learning EMEA; 4th edition (August 9, 2019);
6. Paul Ammann, Jeff Offutt, Introduction to Software Testing 2nd Edition, Cambridge University Press; 2 edition (December 13, 2016);
7. Glenford J. Myers, Corey Sandler, Tom Badgett, The Art of Software Testing 3rd Edition Wiley; 3 edition (November 8,2011);
8. Marius Tomescu - Note de curs si laborator, platforma SUMS

8.7 Project Outline	Teaching methods	Remarks
8.8 Project Outline		

9. Correlation of course outline with the expectations of the epistemic community, professional associations and representative employers within the field of the program

The content of the discipline is consistent with what is done in other university centers in the country and abroad. In order to better adapt the content of the discipline to the requirements of the labor market, meetings were held both with representatives of the business environment and with mathematics and computer science teachers from pre-university education.

10. Evaluation / Grading (if applicable)

Activity type	Evaluation criteria	Evaluation methods	Percentage of the final grade
10.1. Lecture	**- correctness and completeness of knowledge; logical coherence; - the degree of dissimilation of the specialized language; -** **- criteria aimed at attitudinal aspects: conscientiousness, interest in individual study.**	Oral evaluation (final in the exam session): Free presentation to the student; The evaluation conversation; Oral quiz. Written assessment (during the semester): report. Active participation in courses.	15% 20% 10%
10.2. Seminar			
10.3. Laboratory	**- the ability to operate with assimilated knowledge.** **- the ability to apply in practice.** **- criteria related to attitudinal aspects: conscientiousness, interest in individual study.**	Current written works: assignments, projects. Final written evaluation (in the exam session) Active participation in the laboratory.	10% 35% 10%
10.4. Project			

10.5 Minimal performance standard

Minimum standard (knowledge and skills required for grade 5) Knowledge of the fundamental elements of theory, solving a simple application. The final mark is calculated as a weighted average of the marks awarded for the components specified in 10.1 and 10.3. The exam is considered passed if each of the grades 10.1 and 10.3 is at least 5. At each of the exam sessions (including the arrears and increases) the grade is calculated according to the same rule. In the absence/increase session, only the tests for which a passing grade was not obtained (minimum 5) can be taken, unless the student wants to take the tests already passed. Note: Students can participate in the consultation hours (2 hours/week according to the planning established at the beginning of the semester) during which the course and/or seminar/laboratory teacher answers the students' questions and offers additional explanations related to the course content, laboratory applications and assignments.

Course coordinator
Prof. univ. dr.
Valeriu BEIU

Seminar/laboratory/project coordinator
Software developer
Marcela-Florina FLOREA

Head of the Department
Lect. univ. dr.
Lorena Camelia POPA

Dean
Prof. univ dr. Sorin-Florin NĂDĂBAN

MINISTERUL EDUCAȚIEI
UNIVERSITATEA „AUREL VLAICU“ DIN ARAD
310130 Arad, B-dul Revolutiei nr. 77, P.O. BOX 2/158 AR
Tel.: 0040-257- 283010; fax. 0040-257- 280070
http://www.uav.ro; e-mail: rectorat@uav.ro
Operator de date cu caracter personal nr. 2929

SYLLABUS

1. Study programme

1.1. Higher education institution	**„Aurel Vlaicu” University of Arad**
1.2. Faculty	**of Exact Sciences**
1.3. Department	**Department of Mathematics and Computer Science**
1.4. Field of study	**Informatics**
1.5. Study level	**2024-2025**
1.6. Ciclul de studii	**Bachelor**
1.7. Study programme / Qualification	**Computer Science (in English)**
1.8. Form of education	**Full – Time study**

2. Course details

2.1. Name of the course	**GlBF5O02 Security of Information Systems**
2.2. Course coordinator	**Bucerzan Dominic PhD**
2.3. Seminar/laboratory/project coordinator	**Halic Catalin-Raul IT Specialist**
2.4. Study year	**3**
2.5. Semester	**1**
2.6. Evaluation type	**ES**
2.7. Course type	**Compulsory**

3. Estimated total time (hours per semester)

3.1. Hours per week	**4**
3.2. Lecture hours per week	**2**
3.3. Seminar/laboratory/project hours per week	**2**
3.4. Total hours per curriculum	**56**
3.5. Lecture hours per semester	**28**
3.6. Seminar/laboratory/project hours per semester	**28**
	Time division [hrs]
3.4.1. Independent study from textbooks, course support, bibliography and notes	**69**
3.4.2. Additional reading (libraries, specialized electronic platforms and field research)	**30**
3.4.3. Preparing of seminars/laboratories/projects, homework, papers, portfolios and essays	**35**
3.4.4. Tutorial coaching	**0**
3.4.5. Examinations	**4**
3.4.6. Other activities	**0**
3.7. Total individual study hours	**69**
3.8. Total hours per semester	**125**
3.9. Number of ECTS credits	**5**

4. Prerequisites (if applicable)

4.1. Curriculum related	
4.2. Competence related	

5. Conditions (if applicable)

5.1. for the lecture	**Classroom equipped with video projector, internet connection, computers and appropiate specific software**
5.2. for the seminar	
5.3. for the laboratory	Laboratory equipped with computers, internet connection and appropiate software
5.4. for the project	

6. Specific educational objectives (competences to be acquired)

6.1. Competenţe profesionale	**C3.Using computer tools in interdisciplinary context;** **C6.Designing and management af computer networks;** **C7. Using modern technologies for information security.**
6.2. Competenţe transversale	**CT1.Applying the rules of organized and efficient work, of responsible attitudes towards teaching-scientific field, to value the own creative potential, while respecting the principles and norms of professional ethics.** **CT2.Efficient conduct of the activities organized in an inter-disciplynary group and developing the personal communication skills, networking and collaboration with various groups;** **CT3.Using of efficient methods and techniques for learning, informing, research and development of the capacity to value knowledge, adapting to the requirements of a dynamic society and communicating in English and in an Internationally widespread language.**

7. Course outcomes (resulting from the specific educational objectives to be acquired)

7.1. General outcomes	**The information security course lays the groundwork for understanding the challenges of 21st century information security technology and specific methods of defending against cyber attacks.**
7.2. Specific outcomes	**After the course, students must know the main dangers related to computer crime as well as defense methods; the main technologies specific to the field will be studied.**

8. Outline (if applicable)

8.1 Lecture Outline	Teaching methods	Remarks
1. Information security in the 21st century. Internet; Security areas and risk mitigation; Methods for ensuring IT information security	Participatory lecture, case study and problematization	
2. Cryptography	interactive exposure, problematization, case study	
3. Detection and prevention tools	interactive exposure, problematization, case study	
4. Security in operating systems. Windows	interactive exposure, problematization, case study	
5. Security in operating systems. . Lynux	interactive exposure, problematization, case study	
6. Security of computer networks	interactive exposure, problematization, case study	
7. E-commerce security	interactive exposure, problematization, case study	

8.2 Lecture References

1. Bucerzan Dominic, Securitatea informatiei economice in retele de calculatoare, Teza de doctorat, ASE Bucuresti
2.Patriciu Victor-Valeriu, Criptografia şi securitatea reţelelor de calculatoare, Ed.Tehnică, 1994
3.Schneier Bruce, Applied Cryptography, John Wiley & Sons, Inc. , 1996
4.Tanenbaum S. Andrew, Computer Networks, Computer Press Agora, 1998
5.http:/www.wikipedia.org
6.http:/www.hackmagedon.com
7.Leliana Valentina Pârvulescu, Igor Vaslav Vitale, Psihologie aplicată în CyberSecurity, Brăila :Editura Sfântul Ioan, 2016
8.Hu Xiong, Zhen Qin, Athanasios V. Vasilakos, Introduction to Certificateless Cryptography, CRC Press, 2016
9. Information Security Management System, ISO 27001.
10. D. Naccache, E. Simion, Cryptography and Information security. Applications, MATRIX ROM, 2011, ISBN 978-973-755-675-2, 107 pages.

8.3 Seminar Outline	Teaching methods	Remarks
8.4 Seminar References		

8.5 Laboratory Outline	Teaching methods	Remarks
1. Information security in the 21st century. Internet; Security areas and risk mitigation; Methods for ensuring IT information security	interactive exposure, problematization, case study	
2. Cryptography	interactive exposure, problematization, case study	
3. Detection and prevention tools	interactive exposure, problematization, case study	
4. Security in operating systems. Windows	interactive exposure, problematization, case study	
5. Security in operating systems. . Lynux	interactive exposure, problematization, case study	
6. Security of computer networks	interactive exposure, problematization, case study	
7. E-commerce security	interactive exposure, problematization, case study	
8.Practices in security management Standards and legislation; case studies	interactive exposure, problematization, case study	

8.6 Laboratory References

1. Bucerzan Dominic, Securitatea informatiei economice in retele de calculatoare, Teza de doctorat, ASE Bucuresti
2.Patriciu Victor-Valeriu, Criptografia şi securitatea reţelelor de calculatoare, Ed.Tehnică, 1994
3.Schneier Bruce, Applied Cryptography, John Wiley & Sons, Inc. , 1996
4.Tanenbaum S. Andrew, Computer Networks, Computer Press Agora, 1998
5.http:/www.wikipedia.org
6.http:/www.hackmagedon.com
7.Leliana Valentina Pârvulescu, Igor Vaslav Vitale, Psihologie aplicată în CyberSecurity, Brăila :Editura Sfântul Ioan, 2016
8.Hu Xiong, Zhen Qin, Athanasios V. Vasilakos, Introduction to Certificateless Cryptography, CRC Press, 2016
9. Information Security Management System, ISO 27001.
10. D. Naccache, E. Simion, Cryptography and Information security. Applications, MATRIX ROM, 2011, ISBN 978-973-755-675-2, 107 pages.

8.7 Project Outline	Teaching methods	Remarks
8.8 Project Outline		

9. Correlation of course outline with the expectations of the epistemic community, professional associations and representative employers within the field of the program

The problem of IT security is acute in this period of the development of modern society; IT criminality is a reality of the Internet and electronic commerce. The content of the discipline provides the necessary knowledge so that future specialists can take security measures using cryptographic techniques that are absolutely necessary in any company.

10. Evaluation / Grading (if applicable)

Activity type	Evaluation criteria	Evaluation methods	Percentage of the final grade
10.1. Lecture	Completeness of knowledge and degree of assimilation	Oral assessment (final exam): - conversation assessment	50%
10.2. Seminar			
10.3. Laboratory	Ability to apply acquired knowledge	Practical work + project	50%
10.4. Project			

10.5 Minimal performance standard

Acquisition of basic theoretical concepts and the ability to apply them in specific projects.

The minimum mark for each test must be 5 (five).

Course coordinator

Bucerzan Dominic PhD

Seminar/laboratory/project coordinator
Halic Catalin-Raul
IT Specialist

Head of the Department
Lect.univ.dr. Lorena Camelia POPA

Dean
Prof.univ.dr. Sorin-Florin NĂDĂBAN

MINISTERUL EDUCAŢIEI
UNIVERSITATEA „AUREL VLAICU“ DIN ARAD
310130 Arad, B-dul Revoluției nr. 77, P.O. BOX 2/158 AR
Tel : 0040-257- 283010; fax. 0040-257- 280070
http://www.uav.ro; e-mail: rectorat@uav.ro
Operator de date cu caracter personal nr.2929

SYLLABUS

1. Study programme

1.1.Higher education institution	**„Aurel Vlaicu” University of Arad**
1.2.Faculty	**of Exact Sciences**
1.3.Department	**Department of Mathematics and Computer Science**
1.4.Field of study	**Informatics**
1.5.Study level	**2024-2025**
1.6.Cicluldestudii	**Bachelor**
1.7.Study programme / Qualification	**Computer Science (in English**
1.8.Form of education	**Full – Time study**

2. Course details

2.1. Name of the course	**Checking and validating software applications**
2.2. Course coordinator	**Conf. Dr.ing. Cornel Barna**
2.3. Seminar/laboratory/project coordinator	**laboratory**
2.4. Study year	**III**
2.5. Semester	**6**
2.6. Evaluation type	**Exam**
2.7. Course type	**mandatory**

3. Estimated total time (hours per semester)

3.1.Hours per week	**4**
3.2.Lecture hours per week	**2**
3.3.Seminar/laboratory/project hours per week	**2**
3.4. Total hours per curriculum	**54**
3.5.Lecture hours per semester	**28**
3.6.Seminar/laboratory/project hours per semester	**28**
	Time division[hrs]
3.4.1.Independent study from textbooks, course support, bibliography and notes	**20**
3.4.2.Additional reading (libraries, specialized electronic platforms and field research)	**19**
3.4.3.Preparing of seminars/laboratories/projects, homework, papers, portfolios and essays	**20**
3.4.4.Tutorial coaching	**5**
3.4.5.Examinations	**3**
3.4.6.Other activities	**2**
3.7. Total individual study hours	**69**
3.8.Total hours per semester	**97**
3.9.Number of ECTS credits	**6**

4. Prerequisites (if applicable)

4.1.Curriculum related	Object oriented programming, Advanced programming methods, Web programming
4.2.Competence related	Average language programming skills

5. Conditions (if applicable)

5.1.for the lecture	Classroom with laptops and video projector
5.2.for the seminar	
5.3.for the laboratory	Laboratory with computers
5.4.for the project	

6. Specific educational objectives (competences to be acquired)

6.1.Competențeprofesionale	**C1. Programarea în limbaje de nivel înalt** **C2. Dezvoltarea și întreținerea aplicațiilor informatice** **C3 Utilizarea instrumentelor informatice în context interdisciplinar** **C4 Utilizarea bazelor teoretice ale informaticii și a modelelor formale** **C5. Proiectarea și gestiunea bazelor de date** **C6. Proiectarea si administrarea rețelelor de calculatoare**
6.2.Competențetransversale	**CT1 Aplicarea regulilor de muncă organizată și eficientă, a unor atitudini responsabile față de domeniul didactic-științific, pentru valorificarea creativă a propriului potențial, cu respectarea principiilor și a normelor de etică profesională;** **CT2 Desfășurarea eficientă a activităților organizate într-un grup inter-disciplinar și dezvoltarea capacităților empatice de comunicare inter-personală, de relaționare și colaborare cu grupuri diverse;** **CT3 Utilizarea unor metode și tehnici eficiente de învățare, informare, cercetare și dezvoltare a capacităților de valorificare a cunoștințelor, de adaptare la cerințele unei societăți dinamice și de comunicare într-o limbă de circulație internațională.**

7. Course outcomes (resulting from the specific educational objectives to be acquired)

7.1. General outcomes	Understanding the notions of partial and totally correct algorithm; Formation of algorithm design skills in parallel with demonstrating their correctness; Knowledge of software testing and verification methods; Training in the design of the correct programs from the specifications; Forming a modern style of programming.
7.2.Specific outcomes	Students will know how it works and what are the steps of an inspection, either of the source code or of the specification from each stage of software development. Students will know how to provide from the specification and design phase the creation of test cases to help them develop a more robust software system. Students will know how to use the tools to manage the testing process. Students will know how to design test cases using different criteria.

8. Outline(if applicable)

8.1Lecture Outline	Teaching methods	Remarks
1. Introductory notions. Short history of software systems verification and validation 2. Getting started. Presentation of general topics and notions 3. Verification and validation of software systems. Program inspection 4. Program testing (1): The concept of program testing. Test criteria. Blackbox testing. 5. Program testing (2): Test criteria. White-box testing. 6. Test levels. Types of testing 7. Testing Web applications. Selenium Web Driver 8. Symbolic execution 9. Checking the models 10 Theory of program correctness (I) The evolution of the concept of fairness. Floyd and Hoare's contributions 11 Theory of program correctness (II) Dijkstra's contributions 12 Quality assurance of software products. Quality control 13 Testing skills and tester ability 14 Summary of the topics presented	Presentation, description, explanations, examples, dialogue	2 hours/topics

8.2Lecture References

1.R. Pressman, Software engineering: a practinioner's approach,7-th edition, Higher Education, 2010
2. Crispin, Grecory, Agile testing: a practical guide for testers and agile teams, Addison-Wesley, 2009
3. M. Pezzand, M. Young, Software Testing and Analysis: Process, Principles and Techniques, John Wiley & Sons, 2008
4. K. Naik, P. Tripathy, Software testing and quality assurance. Theory and Practice, A John Wiley & Sons, Inc., 2008
5. J. P. Katoen, Principles of Model Checking, MIT Press, May 2008
6. R. Patton, Software Testing, Sams Publishing, 2005
7. Glenford J. Myers, The Art of Software Testing, John Wiley & Sons, Inc., 2004

8.3Seminar Outline	Teaching methods	Remarks

8.4SeminarReferences

8.5Laboratory Outline	Teaching methods	Remarks
1 -14. Practical aspects based on the topics discussed in the course	Computer example.Functionality testing	28 hours

8.6Laboratory References

1.R. Pressman, Software engineering: a practinioner's approach,7-th edition, Higher Education, 2010

2. Crispin, Grecory, Agile testing: a practical guide for testers and agile teams, Addison-Wesley, 2009
3. M. Pezzand, M. Young, Software Testing and Analysis: Process, Principles and Techniques, John Wiley & Sons, 2008
4. K. Naik, P. Tripathy, Software testing and quality assurance. Theory and Practice, A John Wiley & Sons, Inc., 2008
5. J. P. Katoen, Principles of Model Checking, MIT Press, May 2008
6. R. Patton, Software Testing, Sams Publishing, 2005
7. Glenford J. Myers, The Art of Software Testing, John Wiley & Sons, Inc., 2004

8.7Project Outline	Teaching methods	Remarks
8.8Project Outline		

9. Correlation of course outline with the expectations of the epistemic community, professional associations and representative employers within the field of the program

10. Evaluation / Grading(if applicable)

Activity type	Evaluation criteria	Evaluation methods	Percentage of the final grade
10.1.Lecture	Knowledge Understanding	Written paper	60%
10.2. Seminar			
10.3. Laboratory	Knowledge and understanding; - Ability to explain and interpret; - Complete and correct solution of the requirements.	- Certified application activity / laboratory / practical works - Tests during the semester	40%
10.4.Project			

10.5Minimal performance standard
1. The student knows the main concepts, recognizes them, defines them correctly and builds a simple application;
2. The specialized language is simple, but used correctly;
3. Minimum grade 5 in the laboratory;
4. To solve well a minimum of topics - questions and applications.

Course coordinator
Conf.univ.drBarna Cornel

Seminar/laboratory/project coordinator
Conf.univ.drBarna Cornel

Head of the Department
Lector dr.Popa Lorena

Dean
Prof.univ.dr.Nadaban Sorin

MINISTERUL EDUCAȚIEI
UNIVERSITATEA „AUREL VLAICU“ DIN ARAD
310130 Arad, B-dul Revolutiei nr. 77, P.O. BOX 2/158 AR
Tel.: 0040-257- 283010; fax. 0040-257- 280070
http://www.uav.ro; e-mail: rectorat@uav.ro
Operator de date cu caracter personal nr. 2929

SYLLABUS

1. Study programme

1.1. Higher education institution	**„Aurel Vlaicu” University of Arad**
1.2. Faculty	**of Exact Sciences**
1.3. Department	**Department of Mathematics and Computer Science**
1.4. Field of study	**Informatics**
1.5. Study level	**2024-2025**
1.6. Ciclul de studii	**Bachelor**
1.7. Study programme / Qualification	**Computer Science (in English)**
1.8. Form of education	**Full – Time study**

2. Course details

2.1. Name of the course	**Cryptography**
2.2. Course coordinator	**Dragoi Vlad-Florin**
2.3. Seminar/laboratory/project coordinator	**Halic Catalin-Raul**
2.4. Study year	**3**
2.5. Semester	**2**
2.6. Evaluation type	**Exam**
2.7. Course type	**Mandatory**

3. Estimated total time (hours per semester)

3.1. Hours per week	**3**
3.2. Lecture hours per week	**2**
3.3. Seminar/laboratory/project hours per week	**1**
3.4. Total hours per curriculum	**42**
3.5. Lecture hours per semester	**28**
3.6. Seminar/laboratory/project hours per semester	**14**
	Time division [hrs]
3.4.1. Independent study from textbooks, course support, bibliography and notes	**15**
3.4.2. Additional reading (libraries, specialized electronic platforms and field research)	**15**
3.4.3. Preparing of seminars/laboratories/projects, homework, papers, portfolios and essays	**24**
3.4.4. Tutorial coaching	
3.4.5. Examinations	**4**
3.4.6. Other activities	
3.7. Total individual study hours	**58**
3.8. Total hours per semester	**100**
3.9. Number of ECTS credits	**4**

4. Prerequisites (if applicable)

4.1. Curriculum related	
4.2. Competence related	Knowledge of a programming language

5. Conditions (if applicable)

5.1. for the lecture	**Lecture room equipped with video projector, internet connection and software tools**
5.2. for the seminar	
5.3. for the laboratory	Laboratory room equipped with networked computers, internet connection and adequate software
5.4. for the project	

6. Specific educational objectives (competences to be acquired)

6.1. Competenţe profesionale	**C1.Programming in high level programming languages;** **C2.Development and maintenance of computer applications;** **C3.Using computer tools in interdisciplinary context;**
6.2. Competenţe transversale	**CT1.Applying the rules of organized and efficient work, of responsible attitudes towards teaching-scientific field, to value the own creative potential, while respecting the principles and norms of professional ethics.** **CT2.Efficient conduct of the activities organized in an inter-disciplynary group and developing the personal communication skills, networking and collaboration with various groups;** **CT3.Using of efficient methods and techniques for learning, informing, research and development of the capacity to value knowledge, adapting to the requirements of a dynamic society and communicating in English and in an Internationally widespread language.**

7. Course outcomes (resulting from the specific educational objectives to be acquired)

7.1. General outcomes	**During this lecture, the students will be familiarized with the main issues and topics in cryptography. All of these will allow them to increase their chances of being hired and integrate multidisciplinary teams.**
7.2. Specific outcomes	**Students will be able to prove that they have acquired knowledge about the main vulnerabilities of an information system, as well as the different techniques employed to defend any such system. They will be able to implement different criptographic protocols starting with Cesar's scheme and going to RSA publick key encryption scheme.**

8. Outline (if applicable)

8.1 Lecture Outline	Teaching methods	Remarks
1. Cybersecurity in the XXI century.	Presentation	4 hours
2. Mathematical tools for cryptography		2hours
3. Classic and modern cryptography. Cryptanalisys.	Lecture using video projector and the internet	4 hours
4. Secret-key cryptography		6 hours
5. Public-key cryptography	Web search	6 hours
6. Digital signatures	Interactive discussions	2 hours
7. E-commerce and security	Examples	2 hours
8. Legislation		2 hours

8.2 Lecture References

1. Bucerzan Dominic, Securitatea informatiei economice in retele de calculatoare, Teza de doctorat, ASE Bucuresti, 2005
2. Patriciu Victor-Valeriu, Criptografia şi securitatea reţelelor de calculatoare, Ed.Tehnică, 1994
3. Schneier Bruce, Applied Cryptography, John Wiley & Sons, Inc. , 1996
4. Eric Charton, Hacker's Guide, 5th edition, Pearson, 2013
5. Adrian Constantin Atanasiu, Securitatea Informatiei vol1 (Criptografie), InfoData Cluj, 2007
6. Adrian Constantin Atanasiu, Securitatea Informatiei vol2 (Protocolae criptografice), InfoData Cluj, 2007
7. Vlad Florin Dragoi, Side-channel attacks în code-based cryptography, Master thesis, 2013, University of Lyon 1, France, [Online] https://docs.google.com/file/d/0B4Cy03-L745ZZ2pXTlpBTWE2MGM/edit
8. Vlad Florin Dragoi, Approche algébrique pour l'étude et la résolution de problèmes algorithmiques issus de la cryptographie et la théorie des codes, PhD Thesis, University of Rouen-Normandy, 2017, [Online] https://hal.archives-ouvertes.fr/tel-01627324/

8.3 Seminar Outline	Teaching methods	Remarks
1. Cybersecurity in the XXI century	Case studies	3 hours
2. Mathematical tools for cryptography	Examples	4 hours
3. Secret-key cryptography	Individual study	6 hours
4. Public-key cryptography	Brainstorming	6 hours
5. Digital signatures	Practical challenges	3 hours
6. E-commerce and security		3 hours
7. Legislation		3 hours

8.4 Seminar References

1. Bucerzan Dominic, Securitatea informatiei economice in retele de calculatoare, Teza de doctorat, ASE Bucuresti, 2005
2. Patriciu Victor-Valeriu, Criptografia şi securitatea reţelelor de calculatoare, Ed.Tehnică, 1994
3. Schneier Bruce, Applied Cryptography, John Wiley & Sons, Inc. , 1996
4. Eric Charton, Hacker's Guide, 5th edition, Pearson, 2013
5. Adrian Constantin Atanasiu, Securitatea Informatiei vol1 (Criptografie), InfoData Cluj, 2007
6. Adrian Constantin Atanasiu, Securitatea Informatiei vol2 (Protocolae criptografice), InfoData Cluj, 2007
7. Vlad Florin Dragoi, Side-channel attacks în code-based cryptography, Master thesis, 2013, University of Lyon 1, France, [Online] https://docs.google.com/file/d/0B4Cy03-L745ZZ2pXTlpBTWE2MGM/edit
8. Vlad Florin Dragoi, Approche algébrique pour l'étude et la résolution de problèmes algorithmiques issus de la cryptographie et la théorie des codes, PhD

Thesis, University of Rouen-Normandy, 2017, [Online] https://hal.archives-ouvertes.fr/tel-01627324/

8.5 Laboratory Outline	Teaching methods	Remarks
8.6 Laboratory Outline		
8.7 Project Outline	Teaching methods	Remarks
8.8 Project Outline		

9. Correlation of course outline with the expectations of the epistemic community, professional associations and representative employers within the field of the program

This course is taught in similar programs at many universities, both in Romania and abroad. For a better matching with the demands of the labor market, meetings with employers' representatives, business representatives, and specialty teachers from the pre-university education system have been organized. Using English brings and added value to the program, enabling the hiring of graduates by multinational companies (both abroad and in Romania).

10. Evaluation / Grading (if applicable)

Activity type	Evaluation criteria	Evaluation methods	Percentage of the final grade
10.1. Lecture	□ Level of mastering the domain-specific vocabulary □ Logical consistency □ Extent of correctness and completeness of knowledge □ Responsibility □ Commitment □ Resolution	Final written exam at the end of the semester Active participation	40% 10%
10.2. Seminar	□ Ability to use the knowledge □ Ability to apply theoretical knowledge to practical cases □ Responsibility □ Commitment □ Resolution	Partial written exam during the semester Independent work, homework Active participating	30% 10% 10%
10.3. Laboratory			
10.4. Project			

10.5 Minimal performance standard

Proper mastering of the basics, understanding the fundamental notions/concepts, fluent with the domain-specific vocabulary, and able to analyze and explain simple cases.

Course coordinator
Dragoi Vlad-Florin

Seminar/laboratory/project coordinator
Dragoi Vlad-Florin

Head of the Department
Lector Popa Lorena

Dean
Prof.univ.dr. Nadaban Sorin

MINISTERUL EDUCAŢIEI ȘI CERCETĂRII
UNIVERSITATEA „AUREL VLAICU“ DIN ARAD
310130 Arad, B-dul Revoluției nr. 77, P.O. BOX 2/158 AR
Tel : 0040-257- 283010; fax. 0040-257- 280070
http://www.uav.ro; e-mail: rectorat@uav.ro
Operator de date cu caracter personal nr.2929

SYLLABUS

1. Study programme

1.1. Higher education institution	**„AUREL VLAICU" UNIVERSITY OF ARAD**
1.2.Faculty	**Faculty of Exact Sciences**
1.3. Department	**Department of Mathematics and Computer Science**
1.4. Field of study	**Informatics**
1.5. Academic year	**2024-2025**
1.6. Study level	**Licence**
1.7. Study programme	**Informatics -english**
1.8. Form of education	**Full-time education**

2. Course details

2.1. Name of the course	Human-computer interface
2.2. Course coordinator	
2.3. Seminar/laboratory/project coordinator	laboratory
2.4. Study year	III
2.5. Semester	6
2.6. Evaluation type	Exam
2.7. Course type	mandatory

3. Estimated total time (hours per semester)

3.1. Hours per week	4
3.2. Lecture hours per week	2
3.3. Seminar/laboratory/project hours per week	2
3.4. Total hours per curriculum	56
3.5. Lecture hours per curriculum	28
3.6. Seminar/laboratory/project hours per curriculum	28
Time division [Hours]	
3.4.1. Independent study from textbooks, course support, bibliography and notes	30

3.4.2. Additional reading	20
3.4.3. Preparing of seminars/laboratories/projects, homework, papers, portfolios and essay	30
3.4.4. Tutorial coaching	5
3.4.5. Examinations	3
3.4.6. Other activities	6
3.7. Total individual study hours	94
3.8. Total hours per semester	150
3.9. Number of ECTS credits	6

4. Prerequisites (if applicable)

4.1. Curriculum related	Notions of electronics and programming
4.2. Competence related	Average language programming skills

5. Conditions (if applicable)

5.1. Conditions for the lecture	Classroom with laptops and video projector
5.2. Conditions for the seminar	
5.3. Conditions for the laboratory	Laboratory with computers
5.4. Conditions for the project	

6. Specific educational objectives (competences to be acquired)

6.1. Professional competences	C1. Notions of electronics C2. Development of technical skills C3 Use of computer tools in an interdisciplinary context C4 Use of the theoretical bases of computer science C5. Interface design C6. Designing adjacent programs
6.2. Transversal competences	CT1 The application of the rules of organized and efficient work, of responsible attitudes towards the didactic-scientific field, for the creative exploitation of one's own potential, in compliance with the principles and norms of professional ethics; CT2 Effectively carrying out activities organized in an inter-disciplinary group and developing empathetic capacities for inter-personal communication, relating and collaborating with diverse groups; CT3 The use of effective methods and techniques of learning, information, research and development of the capacities to capitalize on knowledge, to adapt to the requirements of a dynamic society and to communicate in an internationally used language.

7. Course outcomes (resulting from the specific educational objectives to be acquired)

7.1. General outcomes	Understanding the notions of human-computer interfaces; Training the skills of designing the hard components of the interfaces; Knowledge of software design methods for interfaces; Training skills to use human-computer interfaces;
7.2.Specific outcomes	Students will know what are the steps of designing an interface Students will know how to foresee the creation of human-computer interfaces from the specification and design phase Students will know how to use the tools for programming an interface. Students will know how to design complex hardware and software units for human-computer interfaces.

8. Course outline

8.1 Lecture	Teaching methods	Remarks
1. Introduction. A brief history of human-computer interfaces 2. Notions about perceptual interfaces 3. Biometric interfaces. 4. Hardware design techniques 5. Design of decoders and multiplexers 6. Types of digital communications used in human-computer interfaces 7. Design of input-output interfaces 8. Driver design 9. Programming human-computer UI interfaces 10. Design of systems with video systems 11. Programming in OpenCV 12. Image processing 13. Recognition of faces and movements in human-computer interfaces 14. Recapitulation in synthesis of the exposed themes	Presentation, description, explanations, examples, dialogue	2 hours 2 hours 2 hours 2 hours 2 hours 2 hours 2 hours 2 hours 2 hours 2 hours 2 hours 2 hours 2 hours
8.2 Lecture references 1. J.Howse OpenCV Computer Vision with Python Ed. Packt Publishing Ltd 2013 2. Image Processing M.Petrou, C.Petrou Ed. John Wiley Ltd 2010 3 J.M.Hugbes Real Word Instrumentation Ed.O Reilly Media Inc 2011 4. R.Lyons Understanding Digital Signal Processing Editura Prentice Hall 2007		
8.3 Seminar	Teaching methods	Remarks
8.4 Seminarreferences		
8.5 / Laboratory	Teaching methods	Remarks
1 -14. Practical aspects based on the topics discussed in the course	Computer example. Functionality testing	28 hours
8.6 Laboratory references 1. J.Howse OpenCV Computer Vision with Python Ed. Packt Publishing Ltd 2013 2. Image Processing M.Petrou, C.Petrou Ed. John Wiley Ltd 2010 3 J.M.Hugbes Real Word Instrumentation Ed.O Reilly Media Inc 2011 4. R.Lyons Understanding Digital Signal Processing Editura Prentice Hall 2007		
8.7 / Project	Teaching methods	Remarks
8.8 Project References		

9. Corroboration / validation of course putline(if applicable)

10. Evaluation / Grading

Activitytype	Evaluation criteria	Evaluation methods	Percentage of the final grade
10.1. Lecture	Knowledge Understanding	Written paper	60%
10.2. Seminar			
10.3. Laboratory	Knowledge and understanding; - Ability to explain and interpret; - Complete and correct solution of the requirements.	- Certified application activity / laboratory / practical works - Tests during the semester	40%
10.4. Project			%

10.5 Minimal performance standard

1. The student knows the main concepts, recognizes them, defines them correctly and builds a simple application;
2. The specialized language is simple, but used correctly;
3. Minimum grade 5 in the laboratory;
4. To solve well a minimum of topics - questions and applications.

Titular	Asistent	DIRECTOR DEPARTAMENT	DECAN
Conf.dr.Barna Cornel	Conf.dr.Barna Corne	Lect. dr. Lorena Popa	Prof. dr. Nadaben Sorin

MINISTERUL EDUCAȚIEI
UNIVERSITATEA „AUREL VLAICU“ DIN ARAD
310130 Arad, B-dul Revolutiei nr. 77, P.O. BOX 2/158 AR
Tel.: 0040-257- 283010; fax. 0040-257- 280070
http://www.uav.ro; e-mail: rectorat@uav.ro
Operator de date cu caracter personal nr. 2929

SYLLABUS

1. Study programme

1.1. Higher education institution	**„Aurel Vlaicu” University of Arad**
1.2. Faculty	**of Exact Sciences**
1.3. Department	**Department of Mathematics and Computer Science**
1.4. Field of study	**Informatics**
1.5. Academic year	**2024-2025**
1.6. Study level	**Bachelor**
1.7. Study programme / Qualification	**Computer Science (in English)**
1.8. Form of education	**Full – Time study**

2. Course details

2.1. Name of the course	**Elaboration of the bachelor's thesis**
2.2. Course coordinator	**PhD. Mihiț Claudia-Luminița**
2.3. Seminar/laboratory/project coordinator	**PhD. Mihiț Claudia-Luminița**
2.4. Study year	**3**
2.5. Semester	**2**
2.6. Evaluation type	**summative**
2.7. Course type	**compulsory**

3. Estimated total time (hours per semester)

3.1. Hours per week	**6**
3.2. Lecture hours per week	**0**
3.3. Seminar/laboratory/project hours per week	**6**
3.4. Total hours per curriculum	**84**
3.5. Lecture hours per semester	**0**
3.6. Seminar/laboratory/project hours per semester	**84**
Time division [hrs]	
3.4.1. Independent study from textbooks, course support, bibliography and notes	**20**
3.4.2. Additional reading (libraries, specialized electronic platforms and field research)	**17**
3.4.3. Preparing of seminars/laboratories/projects, homework, papers, portfolios and essays	**0**
3.4.4. Tutorial coaching	**0**
3.4.5. Examinations	**4**
3.4.6. Other activities	**0**
3.7. Total individual study hours	**41**
3.8. Total hours per semester	**125**
3.9. Number of ECTS credits	**5**

4. Prerequisites (if applicable)

4.1. Curriculum related	Completion of the mandatory fundamental, specialization and complementary subjects provided in the education plan.
4.2. Competence related	Skills of analysis and synthesis of knowledge in the field of undergraduate thesis.

5. Conditions (if applicable)

5.1. for the lecture	
5.2. for the seminar	
5.3. for the laboratory	The laboratories are conducted in the form of meetings between the student and the coordinator of the bachelor's thesis.
5.4. for the project	

6. Specific educational objectives (competences to be acquired)

6.1. Professional skills	**C1. Programming in high level programming languages;** **C2. Development and maintenance of computer applications;** **C3. Using computer tools in interdisciplinary context;** **C4. Using the theoretical bases of computers and formal models;** **C5. Database design and database management;** **C6. Designing and management af computer networks;**
6.2. Transversal skills	**CT1.Applying the rules of organized and efficient work, of responsible attitudes towards teaching-scientific field, to value the own creative potential, while respecting the principles and norms of professional ethics.** **CT3.Using of efficient methods and techniques for learning, informing, research and development of the capacity to value knowledge, adapting to the requirements of a dynamic society and communicating in English and in an Internationally widespread language.**

7. Course outcomes (resulting from the specific educational objectives to be acquired)

7.1. General outcomes	**Synthesizing the information accumulated during studies and the ability to apply them in practice in the form of a bachelor's thesis.**
7.2. Specific outcomes	**1. Familiarizing students with the substantive requirements of a bachelor's thesis** **2. Guiding the students in the elaboration of a paper containing a theoretical and an applied part; which is innovative, interdisciplinary and original.** **3. Monitoring the correct application of the specific methods of analysis in the field of the bachelor's thesis and the observance of the model agreed at the university level for the elaboration of the bachelor's thesis.**

8. Outline (if applicable)

8.1 Lecture Outline	Teaching methods	Remarks
8.2 Lecture References		

8.3 Seminar Outline	Teaching methods	Remarks
8.4 Seminar References		

8.5 Laboratory Outline	Teaching methods	Remarks
1. Establishing the thematic universe of the undergraduate theses 2. Establishing the indicative title, structure and bibliography of the work as a result of the study of the specialized literature 3. Establishing the calendar for the achievement of the undergraduate thesis 4. Discussions on how to prepare the undergraduate thesis: structure the work, technical editing conditions of the work, use of bibliographic references, use of figures, graphs, etc. 5. Discussions regarding the theoretical and methodological aspects of the work according to the chosen theme. 6. Coordination of the application part of the undergraduate work and correct setting of the conclusions 7. Finalization of the undergraduate work and anti-plagiarism check of each work 8. Preparation of the presentation for public support.	Individual and/or group discussions, independent reading and consultations; individual work.	The assignments are proposed by the teaching staff of the Mathematics and Computer Science Department, but can be completed with student proposals. The display of undergraduate assignments is done on the notice board, respectively on the faculty's website until Oct. 15.
8.6 Laboratory References In addition to the bibliography recommended by the coordinating teaching staff and the one chosen by the student, it is recommended to take into account the guide for preparing undergraduate theses, approved at the university level http://www.uav.ro/ro/academic/finalizare-studii		

8.7 Project Outline	Teaching methods	Remarks
8.8 Project Outline		

9. Correlation of course outline with the expectations of the epistemic community, professional associations and representative employers within the field of the program

The content of the discipline is consistent with that of similar subjects taught in university centers in the country and abroad and ensures the methodological universe for students in order to prepare and support the bachelor's thesis.

10. Evaluation / Grading (if applicable)

Activity type	Evaluation criteria	Evaluation methods	Percentage of the final grade
10.1. Lecture			
10.2. Seminar			
10.3. Laboratory	**-choosing the theme and proof of studying the literature of appropriate specialty** **- completing the bachelor's thesis (content plus form)**	-the motivation of the chosen topic, the bibliographic additions made to the bibliography proposed by the coordinator supported by a consistent synthesis of the studied material - the chosen methodology is consistent with achieving the objectives - the work is consistent and well structured - the conclusions are logical and relevant to the theme of the paper - they respect the model agreed at the university level	-motivation of the theme 30% - elaboration of the work 70%
10.4. Project			
10.5 Minimal performance standard **Average level of theoretical knowledge specific to the field and the ability to make a standard application.**			

Course coordinator
Lect.univ.dr. Claudia MIHIȚ

Seminar/laboratory/project coordinator
Lect.univ.dr. Claudia MIHIȚ

Head of the Department
Lect.univ.dr. Lorena Camelia POPA

Dean
Prof.univ.dr. Sorin-Florin NĂDĂBAN

MINISTERUL EDUCAȚIEI
UNIVERSITATEA „AUREL VLAICU“ DIN ARAD
310130 Arad, B-dul Revolutiei nr. 77, P.O. BOX 2/158 AR
Tel.: 0040-257- 283010; fax. 0040-257- 280070
http://www.uav.ro; e-mail: rectorat@uav.ro
Operator de date cu caracter personal nr. 2929

SYLLABUS

1. Study programme

1.1. Higher education institution	**„Aurel Vlaicu” University of Arad**
1.2. Faculty	**of Exact Sciences**
1.3. Department	**Department of Mathematics and Computer Science**
1.4. Field of study	**Informatics**
1.5. Academic year	**2024-2025**
1.6. Study level	**Bachelor**
1.7. Study programme / Qualification	**Computer Science (in English)**
1.8. Form of education	**Full – Time study**

2. Course details

2.1. Name of the course	**Operational research**
2.2. Course coordinator	**PhD. Mihiț Claudia-Luminița**
2.3. Seminar/laboratory/project coordinator	**PhD. Mihiț Claudia-Luminița**
2.4. Study year	**3**
2.5. Semester	**1**
2.6. Evaluation type	**summative**
2.7. Course type	**optional**

3. Estimated total time (hours per semester)

3.1. Hours per week	**3**
3.2. Lecture hours per week	**2**
3.3. Seminar/laboratory/project hours per week	**1**
3.4. Total hours per curriculum	**42**
3.5. Lecture hours per semester	**28**
3.6. Seminar/laboratory/project hours per semester	**14**
	Time division [hrs]
3.4.1. Independent study from textbooks, course support, bibliography and notes	**30**
3.4.2. Additional reading (libraries, specialized electronic platforms and field research)	**25**
3.4.3. Preparing of seminars/laboratories/projects, homework, papers, portfolios and essays	**20**
3.4.4. Tutorial coaching	**4**
3.4.5. Examinations	**4**
3.4.6. Other activities	**0**
3.7. Total individual study hours	**83**
3.8. Total hours per semester	**125**
3.9. Number of ECTS credits	**5**

4. Prerequisites (if applicable)

4.1. Curriculum related	
4.2. Competence related	

5. Conditions (if applicable)

5.1. for the lecture	Internet access The classroom is equipped with a blackboard Computer/Laptop and Video projector
5.2. for the seminar	Internet access Specific equipment and apparatus Blackboard
5.3. for the laboratory	
5.4. for the project	

6. Specific educational objectives (competences to be acquired)

6.1. Professional skills	**C3.Using computer tools in interdisciplinary context;** **C4.Using the theoretical bases of computers and formal models.**
6.2. Transversal skills	**CT1.Applying the rules of organized and efficient work, of responsible attitudes towards teaching-scientific field, to value the own creative potential, while respecting the principles and norms of professional ethics.** **CT3.Using of efficient methods and techniques for learning, informing, research and development of the capacity to value knowledge, adapting to the requirements of a dynamic society and communicating in English and in an Internationally widespread language.**

7. Course outcomes (resulting from the specific educational objectives to be acquired)

7.1. General outcomes	- The student should develop the skills to correctly apply the accumulated knowledge to solve different types of problems. - The student must train and develop his ability to think and analyze operations research problems.
7.2. Specific outcomes	- The student is able to demonstrate that he has acquired sufficient knowledge to understand the basic notions. - The student is able to recognize the main classes/types of operations research problems and select the appropriate methods and techniques for solving them. - The student can create projects for the mathematical modeling of concrete problems.

8. Outline (if applicable)

8.1 Lecture Outline	Teaching methods	Remarks
1. Linear programming	Participatory lecture, problematization, demonstration, exemplification	4 hours
2. The primal SIMPLEX algorithm	Participatory lecture, problematization, demonstration, exemplification	4 hours
3. Duality	Participatory lecture, problematization, demonstration, exemplification	4 hours
4 The dual SIMPLEX algorithm	Participatory lecture, problematization, demonstration, exemplification	4 hours
5. Penalty method	Participatory lecture, problematization, demonstration, exemplification	4 hours
6. The theory of potentials	Participatory lecture, problematization, demonstration, exemplification	4 hours
7. Game theory	Participatory lecture, problematization, demonstration, exemplification	4 hours

8.2 Lecture References

1. **B. E. Breckner, N. Popovici, Probleme de cercetare operațională, Ed. Fundației pentru Studii Europene, 2006.**
2. **W. W. Breckner, Cercetare operationala, Cluj-Napoca, Universitatea "Babes-Bolyai", 1981.**
3. **E. K. P. Chong, S. H. Żak, An Introduction to Optimization, Wiley-Interscience, 2008.**
4. **S. I. Gass, Linear Programming, Dover Publications Inc., 2010.**
5. **C. L. Mihiț, Lecture notes and laboratory-Operational research, SUMS, 2024.**
6. **A. Nemirovski, Introduction to Linear Optimization, WORLD SCIENTIFIC PUB CO INC, 2024.**
7. **D. Opriș, G. Silberberg, Optimizări liniare, discrete, convexe, Ed. Mirton, Timișoara, 1999.**

8. P. Pedregal, Introduction to Optimization, Springer, 2004.
9. Zs. Szabo, Cercetări operaţionale, Ed. Universităţii Petru Maior, Tg.Mureş, 2005.

8.3 Seminar Outline	Teaching methods	Remarks

8.4 Seminar References

8.5 Laboratory Outline	Teaching methods	Remarks
1. Linear programming	Applications, debates	2 hours
2. The primal SIMPLEX algorithm	Applications, debates	2 hours
3. Duality	Applications, debates	2 hours
4 The dual SIMPLEX algorithm	Applications, debates	2 hours
5. Penalty method	Applications, debates	2 hours
6. The theory of potentials	Applications, debates	2 hours
7. Game theory	Applications, debates	2 hours

8.6 Laboratory References

1. B. E. Breckner, N. Popovici, Probleme de cercetare operaţională, Ed. Fundaţiei pentru Studii Europene, 2006.
2. W. W. Breckner, Cercetare operationala, Cluj-Napoca, Universitatea "Babes-Bolyai", 1981.
3. E. K. P. Chong, S. H. Żak, An Introduction to Optimization, Wiley-Interscience, 2008.
4. S. I. Gass, Linear Programming, Dover Publications Inc., 2010.
5. C. L. Mihiţ, Lecture notes and laboratory-Operational research, SUMS, 2024.
6. A. Nemirovski, Introduction to Linear Optimization, WORLD SCIENTIFIC PUB CO INC, 2024.
7. D. Opriş, G. Silberberg, Optimizări liniare, discrete, convexe, Ed. Mirton, Timişoara, 1999.
8. P. Pedregal, Introduction to Optimization, Springer, 2004.
9. Zs. Szabo, Cercetări operaţionale, Ed. Universităţii Petru Maior, Tg.Mureş, 2005.

8.7 Project Outline	Teaching methods	Remarks
8.8 Project Outline		

9. Correlation of course outline with the expectations of the epistemic community, professional associations and representative employers within the field of the program

The content of the discipline is consistent with what is done in other university centers in the country and abroad. In order to better adapt the content of the discipline to the requirements of the labor market, meetings were held with representatives of employers.

10. Evaluation / Grading (if applicable)

Activity type	Evaluation criteria	Evaluation methods	Percentage of the final grade
10.1. Lecture	**• completeness of knowledge;** **• logical coherence;** **• degree of assimilation of the specialized language;** **• the criteria for attitudinal aspects: seriousness, interest in the topic addressed**	oral assessment: presentation of a final project. active participation in courses	30% 10%
10.2. Seminar			
10.3. Laboratory	**• the ability to operate with assimilated knowledge;** **• the ability to apply in practice;** **• conscientiousness and interest in studying.**	making and presenting the final project; homework, projects; active participation in laboratory applications	30% 20% 10%
10.4. Project			

10.5 Minimal performance standard

Average level of theoretical knowledge specific to the field and the ability to make a standard application.

Course coordinator
Lect.univ.dr. Claudia MIHIŢ

Seminar/laboratory/project coordinator
Lect.univ.dr. Claudia MIHIŢ

Head of the Department
Lect.univ.dr. Lorena Camelia POPA

Dean
Prof.univ.dr. Sorin-Florin NĂDĂBAN

MINISTERUL EDUCAȚIEI
UNIVERSITATEA „AUREL VLAICU“ DIN ARAD
310130 Arad, B-dul Revolutiei nr. 77, P.O. BOX 2/158 AR
Tel.: 0040-257- 283010; fax. 0040-257- 280070
http://www.uav.ro; e-mail: rectorat@uav.ro
Operator de date cu caracter personal nr. 2929

SYLLABUS

1. Study programme

1.1. Higher education institution	**„Aurel Vlaicu” University of Arad**
1.2. Faculty	**of Exact Sciences**
1.3. Department	**Department of Mathematics and Computer Science**
1.4. Field of study	**Informatics**
1.5. Study level	**2024-2025**
1.6. Ciclul de studii	**Bachelor**
1.7. Study programme / Qualification	**Computer Science (in English)**
1.8. Form of education	**Full – Time study**

2. Course details

2.1. Name of the course	**Computational geometry**
2.2. Course coordinator	**PhD. Moț Ghiocel**
2.3. Seminar/laboratory/project coordinator	**PhD. Moț Ghiocel**
2.4. Study year	**3**
2.5. Semester	**1**
2.6. Evaluation type	**summative**
2.7. Course type	**optional**

3. Estimated total time (hours per semester)

3.1. Hours per week	**3**
3.2. Lecture hours per week	**2**
3.3. Seminar/laboratory/project hours per week	**1**
3.4. Total hours per curriculum	**42**
3.5. Lecture hours per semester	**28**
3.6. Seminar/laboratory/project hours per semester	**14**
	Time division [hrs]
3.4.1. Independent study from textbooks, course support, bibliography and notes	**30**
3.4.2. Additional reading (libraries, specialized electronic platforms and field research)	**20**
3.4.3. Preparing of seminars/laboratories/projects, homework, papers, portfolios and essays	**25**
3.4.4. Tutorial coaching	**4**
3.4.5. Examinations	**4**
3.4.6. Other activities	**0**
3.7. Total individual study hours	**83**
3.8. Total hours per semester	**125**
3.9. Number of ECTS credits	**5**

4. Prerequisites (if applicable)

4.1. Curriculum related	
4.2. Competence related	

5. Conditions (if applicable)

5.1. for the lecture	Internet access The classroom is equipped with a blackboard Computer/Laptop and Video projector
5.2. for the seminar	Internet access Specific equipment and apparatus Blackboard
5.3. for the laboratory	
5.4. for the project	

6. Specific educational objectives (competences to be acquired)

6.1. Professional skills	**C3.Using computer tools in interdisciplinary context;** **C4.Using the theoretical bases of computers and formal models.**
6.2. Transversal skills	**CT1.Applying the rules of organized and efficient work, of responsible attitudes towards teaching-scientific field, to value the own creative potential, while respecting the principles and norms of professional ethics.** **CT3.Using of efficient methods and techniques for learning, informing, research and development of the capacity to value knowledge, adapting to the requirements of a dynamic society and communicating in English and in an Internationally widespread language.**

7. Course outcomes (resulting from the specific educational objectives to be acquired)

7.1. General outcomes	**- The student should develop his skills to correctly apply the accumulated knowledge to solve different types of problems.** **- The student must train and develop his analysis capacity for computational geometry problems.**
7.2. Specific outcomes	**- The student is able to correctly apply the basic methods and principles in solving practical problems.** **-The student is able to recognize the main classes/types of computational geometry problems and apply the appropriate methods and techniques to solve them.** **- The student can mathematically model some concrete problems.**

8. Outline (if applicable)

8.1 Lecture Outline	Teaching methods	Remarks
1. Computational geometry of the plane	Participatory lecture, problematization, demonstration, exemplification	4 hours
2. Computational geometry of lines in the plane	Participatory lecture, problematization, demonstration, exemplification	4 hours
3. Polygons	Participatory lecture, problematization, demonstration, exemplification	4 hours
4. Computational geometry of conics	Participatory lecture, problematization, demonstration, exemplification	4 hours
5. Approximation of functions	Participatory lecture, problematization, demonstration, exemplification	4 hours
6. Approximation of plane curves	Participatory lecture, problematization, demonstration, exemplification	4 hours
7. Digital curves	Participatory lecture, problematization, demonstration, exemplification	4 hours

8.2 Lecture References

1. M. de Berg, M. van Kreveld, M. Overmars, O. Schwarzkopf, Computational Geometry, Algorithms and Applications, Springer, 2000.
2. G. Cristescu, Geometrie analitică, diferențială și computațională, Ed. Mirton, Timișoara, 2010.
3. S. Devadoss, J. O'Rourke, Discrete and Computational Geometry, Princeton University Press, 2011.
4. J. E. Goodman, Joseph O'Rourke, and Csaba D. Tóth (editors), Handbook of Discrete and Computational Geometry, 3rd edition, CRC Press, 2017.
5. A. Gray, Modern Differential Geometry of Curves and Surfaces with Mathematica, CRC Press, 1999.
6. G. Moț, L. Popa, Algebră liniară. Geometrie analitică și diferențială. Ed. Univ. "Aurel Vlaicu" Arad, 2014.
7. G. Moț, Lecture notes and laboratory-Computational geometry, SUMS, 2024.

8.3 Seminar Outline	Teaching methods	Remarks

8.4 Seminar References		
8.5 Laboratory Outline	Teaching methods	Remarks
1. Computational geometry of the plane	Applications, debates, projects	2 hours
2. Computational geometry of lines in the plane	Applications, debates, projects	2 hours
3. Polygons	Applications, debates, projects	2 hours
4. Computational geometry of conics	Applications, debates, projects	2 hours
5. Approximation of functions	Applications, debates, projects	2 hours
6. Approximation of plane curves	Applications, debates, projects	2 hours
7. Digital curves	Applications, debates, projects	2 hours

8.6 Laboratory References

1. M. de Berg, M. van Kreveld, M. Overmars, O. Schwarzkopf, Computational Geometry, Algorithms and Applications, Springer, 2000.
2. G. Cristescu, Geometrie analitică, diferențială și computațională, Ed. Mirton, Timișoara, 2010.
3. S. Devadoss, J. O'Rourke, Discrete and Computational Geometry, Princeton University Press, 2011.
4. J. E. Goodman, Joseph O'Rourke, and Csaba D. Tóth (editors), Handbook of Discrete and Computational Geometry, 3rd edition, CRC Press, 2017.
5. A. Gray, Modern Differential Geometry of Curves and Surfaces with Mathematica, CRC Press, 1999.
6. G. Moț, L. Popa, Algebră liniară. Geometrie analitică și diferențială. Ed. Univ. "Aurel Vlaicu" Arad, 2014.
7. G. Moț, Lecture notes and laboratory-Computational geometry, SUMS, 2024.

8.7 Project Outline	Teaching methods	Remarks
8.8 Project Outline		

9. Correlation of course outline with the expectations of the epistemic community, professional associations and representative employers within the field of the program

The content of the discipline is consistent with what is done in other university centers in the country and abroad. In order to better adapt the content of the discipline to the requirements of the labor market, meetings were held with representatives of employers.

10. Evaluation / Grading (if applicable)

Activity type	Evaluation criteria	Evaluation methods	Percentage of the final grade
10.1. Lecture	• **completeness of knowledge;** • **logical coherence;** • **degree of assimilation of the specialized language;** • **the criteria for attitudinal aspects: seriousness, interest in the topic addressed**	oral assessment: presentation of a final project. active participation in courses	30% 10%
10.2. Seminar			
10.3. Laboratory	• **the ability to operate with assimilated knowledge;** • **the ability to apply in practice;** • **conscientiousness and interest in studying.**	making and presenting the final project; homework, projects; active participation in laboratory applications	30% 20% 10%
10.4. Project			

10.5 Minimal performance standard

Average level of theoretical knowledge specific to the field and the ability to make a standard application.

Course coordinator
Prof.univ.dr. Ghiocel MOȚ

Seminar/laboratory/project coordinator
Prof.univ.dr. Ghiocel MOȚ

Head of the Department
Lect.univ.dr. Lorena Camelia POPA

Dean
Prof.univ.dr. Sorin-Florin NĂDĂBAN

MINISTERUL EDUCAŢIEI
UNIVERSITATEA „AUREL VLAICU“ DIN ARAD
310130 Arad, B-dul Revoluției nr. 77, P.O. BOX 2/158 AR
Tel : 0040-257- 283010; fax. 0040-257- 280070
http://www.uav.ro; e-mail: rectorat@uav.ro
Operator de date cu caracter personal nr.2929

SYLLABUS

1. Study programme

1.1. Higher education institution	**„Aurel Vlaicu” University of Arad**
1.2. Faculty	**of Exact Sciences**
1.3. Department	**Department of Mathematics and Computer Science**
1.4. Field of study	**Informatics**
1.5. Study year	**2023-2024**
1.6. Study level	**Bachelor**
1.7. Study programme / Qualification	**Computer Science (in English)**
1.8. Form of education	**Full – Time study**

2. Course details

2.1. Name of the course	**GIBS5A11 Web Technologies 2**
2.2. Course coordinator	**Bucerzan Dominic PhD**
2.3. Seminar/laboratory/project coordinator	**Halic Catalin-Raul IT Specialist**
2.4. Study year	**3**
2.5. Semester	**1**
2.6. Evaluation type	**ES**
2.7. Course type	**Op**

3. Estimated total time (hours per semester)

3.1. Hours per week	**4**
3.2. Lecture hours per week	**2**
3.3. Seminar/laboratory/project hours per week	**2**
3.4. Total hours per curriculum	**56**
3.5. Lecture hours per semester	**28**
3.6. Seminar/laboratory/project hours per semester	**28**
	Time division [hrs]
3.4.1. Independent study from textbooks, course support, bibliography and notes	**25**
3.4.2. Additional reading (libraries, specialized electronic platforms and field research)	**25**
3.4.3. Preparing of seminars/laboratories/projects, homework, papers, portfolios and essays	**15**
3.4.4. Tutorial coaching	**2**
3.4.5. Examinations	**2**
3.4.6. Other activities	**0**
3.7. Total individual study hours	**69**
3.8. Total hours per semester	**125**
3.9. Number of ECTS credits	**5**

4. Prerequisites (if applicable)

4.1. Curriculum related	Basic knowledge of web technologies.
4.2. Competence related	

5. Conditions (if applicable)

5.1. for the lecture	**Classroom with laptops, intelligent whiteboard, Internet Connection, IDE**
5.2. for the seminar	
5.3. for the laboratory	**Classroom with laptops, intelligent whiteboard, Internet Connection, IDE**
5.4. for the project	

6. Specific educational objectives (competences to be acquired)

6.1. Professional competencies	**C1. Programming in high level programming languages;** **C2. Development and maintenance of computer applications;** **C4. Using the theoretical bases of computers and formal models;**
6.2. Transversal competencies	**CT1. Applying the rules of organized and efficient work, of responsible attitudes towards teaching-scientific field, to value the own creative potential, while respecting the principles and norms of professional ethics.**

7. Course outcomes (resulting from the specific educational objectives to be acquired)

7.1. General outcomes	**Consolidation of general knowledge gained about web pages and technologies. Deepening the concepts of web application development.**
7.2. Specific outcomes	**Students will be able to create a web application using technologies such as: HTML, CSS, JavaScript, TypeScript, React, Angular, but also C#, ASP.NET, Razor.**

8. Outline (if applicable)

8.1 Lecture Outline	Teaching methods	Remarks
Review – WWW, Internet, HTML, CSS, Javascript, C#, SQL Server	Heuristic conversation, exemplification, problematization	6 hours
Javascript, Typescript, React, Angular, Bootstrap	Exemplification, debate	6 hours
ASP.NET Core	Exemplification, heuristic conversation	6 hours
Razor Pages	Example, demonstration	6 hours
MVC, API	Example, demonstration	4 hours

8.2 Lecture References

Books: Professional C# 4 and .NET 4 – C. Nagel, B. Evjen, J. Glynn, K. Watson, M. Skinner
Sites:
- **https://learn.microsoft.com/en-us/dotnet/csharp/tour-of-csharp/tutorials/**
- **https://www.tutorialspoint.com/csharp/index.htm**
- **https://www.tutorialsteacher.com/csharp**
- **https://www.w3schools.com/cs/index.php**
- **https://learn.microsoft.com/en-us/sql/sql-server/tutorials-for-sql-server-2016?view=sql-server-ver16**
- **https://www.w3schools.com/html/**
- **https://www.w3schools.com/css/**
- **https://learn.microsoft.com/en-us/aspnet/core/tutorials/first-mvc-app/start-mvc?view=aspnetcore-6.0&tabs=visual-studio**
- **https://www.tutorialsteacher.com/core**

Youtube:
- **https://www.youtube.com/user/IAmTimCorey**
- **https://www.youtube.com/c/programmingwithmosh**
- **https://www.youtube.com/c/Csharp-video-tutorialsBlogspot**

https://www.youtube.com/watch?v=7GVFYt6_ZFM&list=PL08903FB7ACA1C2FB
Other: PluralSight - https://www.pluralsight.com/product/skills/free
Marius Tomescu - Lecture notes - SUMS, 2023

8.3 Seminar Outline	Teaching methods	Remarks

8.4 Seminar References

8.5 Laboratory Outline	Teaching methods	Remarks
HTML, CSS, Javascript - elements, styles, scripts, examples by applications	Exemplification, Heuristic conversation	4 hours
C#, SQL Server - overview of databases and application types in C#	Exemplification, Heuristic conversation	4 hours
Javascript, Typescript - frameworks: React and Angular - examples of applications	Exemplification, Heuristic conversation, Exercise	4 hours
ASP.Net Core - types of applications	Exemplification	2 hours
Application in ASP.NET Core with Razor pages - BookStore	Exemplification, Heuristic conversation, Exercise	10 hours
Other types of applications: MVC, API	Exemplification	4 hours

8.6 Laboratory bibliography

Books: Professional C# 4 and .NET 4 – C. Nagel, B. Evjen, J. Glynn, K. Watson, M. Skinner
Sites:
• https://learn.microsoft.com/en-us/dotnet/csharp/tour-of-csharp/tutorials/
• https://www.tutorialspoint.com/csharp/index.htm
• https://www.tutorialsteacher.com/csharp
• https://www.w3schools.com/cs/index.php
• https://learn.microsoft.com/en-us/sql/sql-server/tutorials-for-sql-server-2016?view=sql-server-ver16
• https://www.w3schools.com/html/
• https://www.w3schools.com/css/
• https://learn.microsoft.com/en-us/aspnet/core/tutorials/first-mvc-app/start-mvc?view=aspnetcore-6.0&tabs=visual-studio
• https://www.tutorialsteacher.com/core
Youtube:
• https://www.youtube.com/user/IAmTimCorey
• https://www.youtube.com/c/programmingwithmosh
• https://www.youtube.com/c/Csharp-video-tutorialsBlogspot
https://www.youtube.com/watch?v=7GVFYt6_ZFM&list=PL08903FB7ACA1C2FB
Other: PluralSight - https://www.pluralsight.com/product/skills/free
Marius Tomescu - Labs notes - SUMS, 2023

8.7 Project Outline	Teaching methods	Remarks
8.8 Project Outline		

9. Correlation of course outline with the expectations of the epistemic community, professional associations and representative employers within the field of the program

The content of the discipline is consistent with the content of similar disciplines from other university centers in the country and abroad. For a better adaptation to the requirements of the labor market of the content of the discipline, meetings were held both with employers - representatives of the business environment and with software developers.

10. Evaluation / Grading (if applicable)

Activity type	Evaluation criteria	Evaluation methods	Percentage of the final grade
10.1. Lecture	**• correctness and completeness of knowledge** **• logical coherence** **• degree of assimilation of the specific dialect** **• conscientiousness, interest in study**	• Oral assessment • Active participation in courses.	40% 10%
10.2. Seminar			
10.3. Laboratory	**• the ability to operate with assimilated knowledge;** **• the ability to apply in practice** **• conscientiousness, interest in studying**	• Oral evaluation (final in the exam session): • completion and presentation of the final project • Assignments, projects along the way	10% 30% 10%
10.4. Project			

10.5 Minimal performance standard

Solving a simple application.

Course coordinator
Bucerzan Dominic PhD

Seminar/laboratory/project coordinator
Halic Catalin-Raul IT Specialist

Head of the Department
Lecturs PhD. Popa Lorena

Dean
Prof.univ.dr. Sorin-Florin NĂDĂBAN

MINISTERUL EDUCAŢIEI
UNIVERSITATEA „AUREL VLAICU“ DIN ARAD
310130 Arad, B-dul Revoluției nr. 77, P.O. BOX 2/158 AR
Tel : 0040-257- 283010; fax. 0040-257- 280070
http://www.uav.ro; e-mail: rectorat@uav.ro
Operator de date cu caracter personal nr.2929

SYLLABUS

1. Study programme

1.1. Higher education institution	**„Aurel Vlaicu” University of Arad**
1.2. Faculty	**of Exact Sciences**
1.3. Department	**Department of Mathematics and Computer Science**
1.4. Field of study	**Informatics**
1.5. Study level	**2024-2025**
1.6. Ciclul de studii	**Bachelor**
1.7. Study programme / Qualification	**Computer Science (in English**
1.8. Form of education	**Full – Time study**

2. Course details

2.1. Name of the course	**GIBS5A14 Computer Graphics**
2.2. Course coordinator	**dr. Gabor Andrei-Marius**
2.3. Seminar/laboratory/project coordinator	**dr. Gabor Andrei-Marius**
2.4. Study year	**3**
2.5. Semester	**1**
2.6. Evaluation type	**ES**
2.7. Course type	**Op**

3. Estimated total time (hours per semester)

3.1. Hours per week	**4**
3.2. Lecture hours per week	**2**
3.3. Seminar/laboratory/project hours per week	**2**
3.4. Total hours per curriculum	**56**
3.5. Lecture hours per semester	**28**
3.6. Seminar/laboratory/project hours per semester	**28**
Time division [hrs]	
3.4.1. Independent study from textbooks, course support, bibliography and notes	**20**
3.4.2. Additional reading (libraries, specialized electronic platforms and field research)	**10**
3.4.3. Preparing of seminars/laboratories/projects, homework, papers, portfolios and essays	**25**
3.4.4. Tutorial coaching	**8**
3.4.5. Examinations	**4**
3.4.6. Other activities	**2**
3.7. Total individual study hours	**69**
3.8. Total hours per semester	**125**
3.9. Number of ECTS credits	**5**

4. Prerequisites (if applicable)

4.1. Curriculum related	Knowledge and skills from the Data Structures, Procedural Programming, Analytical Geometry, Linear Algebra, Mathematical Analysis courses
4.2. Competence related	Ability and ability to solve problems using the JAVA programming language

5. Conditions (if applicable)

5.1. for the lecture	**Room equipped with video projector and internet connection**
5.2. for the seminar	
5.3. for the laboratory	**IT laboratory equipped with computers connected to the Internet: Licensed operating system and programming environment (free software)**
5.4. for the project	

6. Specific educational objectives (competences to be acquired)

6.1. Competențe profesionale	**C1.Programming in high level programming languages;** **C2.Development and maintenance of computer applications;** **C3.Using computer tools in interdisciplinary context;** **C4.Using the theoretical bases of computers and formal models;** **C5.Database design and database management;** **C6.Designing and management af computer networks;**
6.2. Competențe transversale	**CT1.Applying the rules of organized and efficient work, of responsible attitudes towards teaching-scientific field, to value the own creative potential, while respecting the principles and norms of professional ethics.** **CT2.Efficient conduct of the activities organized in an inter-disciplynary group and developing the personal communication skills, networking and collaboration with various groups;** **CT3.Using of efficient methods and techniques for learning, informing, research and development of the capacity to value knowledge, adapting to the requirements of a dynamic society and communicating in English and in an Internationally widespread language.**

7. Course outcomes (resulting from the specific educational objectives to be acquired)

7.1. General outcomes	**In this discipline, students have the opportunity to acquire the necessary basic knowledge in the field of graphic representations, which find numerous applications in cartography, meteorology, medicine, office automation, advertising, film production, etc.** **KNOWLEDGE: Acquiring the fundamental concepts, principles and basic techniques in the field of computer graphics; Knowledge and use of concepts and techniques for designing, developing and using basic software and application software for computer graphics; Knowledge and use of CG (Computer Graphics) concepts and techniques for solving 2D and 3D modeling and representation problems;** **SKILLS: Learning the general principles of CG and using programming languages in 2D/3D modeling and representation to solve problems** **COMPETENCES: the ability to use CG knowledge and software facilities for CG; through the activities in the laboratory, it is aimed that the students obtain skills regarding the use of computer systems in information technology in the field of computer graphics**
7.2. Specific outcomes	**Knowledge and understanding (knowledge and appropriate use of discipline-specific notions)** **Enriching information technology knowledge by adding new knowledge, new methods and already existing techniques; enriching the language in the field of computers, the correct use of modeling and representation in solving problems** **The development of the synthesis capabilities of some fundamental notions of CG** **Ability to understand CG concepts and techniques and to apply them in concrete cases • Ability to apply CG knowledge and software in solving problems in 2D/3D graphics** **Explanation and interpretation** **Training the ability to design solutions to problems specific to the field of knowledge;** **The use of language specific to the discipline in written and oral communication;** **Instrumental - applications** **Application of research concepts and methods to formulate projects and argue the chosen solutions;** **Elaboration of scientific reports, commentaries and case analyses;** **The use of scientific research methods and procedures to design and write papers in order to participate in scientific competitions;**

8. Outline (if applicable)

8.1 Lecture Outline	Teaching methods	Remarks
Graphics systems **Classification** **Display devices** **Input devices** **Graphic system architectures**	**Lecture accompanied by materials in electronic format (PDF)**	**4 hours**
Computer graphics software **- software oriented to the field, to the user, to the programmer** **- standardization for graphic systems**	**Lecture accompanied by materials in electronic format (PDF)**	**2 hours**
Two-dimensional graphic transformations. translation, **scaling, Rotation** **Composition of transformations** **Inverse geometric transformations** **Transformations of the coordinate system Shearing**	**Lecture accompanied by materials in electronic format (PDF)**	**2 hours**
Parallel projections **Prospective projections**	**Lecture accompanied by materials in electronic format (PDF)**	**2 hours**

Cropping algorithms. Cropping points. Cutting lines The Cohen-Sutherland algorithm	**Lecture accompanied by materials in electronic format (PDF)**	**2 hours**
View transformations 2D View Transformations 3D View Transformations	**Lecture accompanied by materials in electronic format (PDF)**	**2 hours**
Modeling and representation of curves and surfaces - Approximation and generation of curves in 2D graphics (Bezier shapes, B-spline shapes)	**Lecture accompanied by materials in electronic format (PDF)**	**2 hours**
Cohen-Sutherland clipping algorithm (clipping of a segment against a window rectangular)	**Lecture accompanied by materials in electronic format (PDF)**	**2 hours**
The clipping algorithm of a segment against a convex or non-convex polygonal window	**Lecture accompanied by materials in electronic format (PDF)**	**2 hours**
The clipping algorithm of a segment against a convex or non-convex polygonal window - The clipping algorithm (Sutherland-Hodgman) of any polygon compared to a convex window	**Lecture accompanied by materials in electronic format (PDF)**	**2 hours**
Textures. Generalities. Generating textures	**Lecture accompanied by materials in electronic format (PDF)**	**2 hours**
Reflection and lighting patterns	**Lecture accompanied by materials in electronic format (PDF)**	**2 hours**
ConCluSIonS	**Lecture accompanied by materials in electronic format (PDF)**	**2 hours**

8.2 Lecture References

1. Gr. Albeanu, Grafica pe calculator. Algoritmi fundamentali, Editura Universitatii Bucuresti, 2011
2. FL. . Moldoveanu, Grafica pe calculator, TEORA, 1996
3. Calin Marin Vaduva, Programarea in Java, Editura Albastra, Cluj-Napoca, 2012
4. D.Petcu, L.Cucu, Principii ale graficii pe calculator, Editura Excelsior, Timisoara, 1995 (Online Petcu & Cucu)
5. Titus Felix Furtuna, Grafica interactiva cu aplicatii in Java si Java 3D, Editura ASE, Bucuresti, 2017
6. http://www.w3schools.com , accesare 2022
7. http://www.w3.org/Graphics/SVG/ , accesare 2022
8.. J. O'Rourke. Computational Geometry in C, Cambridge Universty Press, 1998

8.3 Seminar Outline	Teaching methods	Remarks

8.4 Seminar References

8.5 Laboratory Outline	Teaching methods	Remarks
Introductory notions. Presentation of works	**Explanations, conversation and testing**	**2 hours**
Introduction to JAVA	**Explanations, conversation and testing**	**2 hours**
2D graphics transformations	**Explanations, conversation and testing**	**4 hours**
Algorithms for generating some geometric shapes	**Explanations, conversation and testing**	**4 hours**
Cropping algorithms	**Explanations, conversation and testing**	**4 hours**
Implementation and testing of algorithms - the raster algorithm (J. Bresenham) for drawing the right segment - test examples	**Explanations, conversation and testing**	**4 hours**
Implementation and testing of algorithms - Cohen-Sutherland clipping algorithm - test examples	**Explanations, conversation and testing**	**4 hours**
Conception, elaboration, implementation of projects - special CG algorithms - examples of problems	**Explanations, conversation and testing**	**2 hours**
Graphical interfaces	**Explanations, conversation and testing**	**2 hours**

8.6 Laboratory Outline

1. Moldoveanu ş.a. - Grafică electronică pe calculator - Editura Teora, Bucureşti, 2016
2. M. Ghinea, V. Zamfir - MATLAB. Calcul numeric. Grafică. Aplicaţii - Editura Teora, Bucureşti, 2018
3. M. Vladu ş.a. - Grafică pe calculator în limbajele PASCAL şi C. Implementare - Editura Tehnică, Bucureşti, 2012
4. M. Vladu ş.a. - Grafică pe calculator în limbajele PASCAL şi C. Aplicaţii - Editura Tehnică, Bucureşti, 2020
5. R.Baciu, D.Volovici – Sisteme de prelucrare grafică – Editura Albastră, Cluj, 2009
6. M. Pater – Elemente de grafică pe calculator – Editura Universităţii din Oradea, ISBN 973-613-203-X, 2012
7. Grava C. – Grafică electronică pe calculator - disponibilă pe pagina web
8. http://cgrava.webhost.uoradea.ro/documentatie_Grafica.html

8.7 Project Outline	Teaching methods	Remarks

8.8 Project Outline

9. Correlation of course outline with the expectations of the epistemic community, professional associations and representative employers within the field of the program

By learning the theoretical-methodological concepts and approaching the practical aspects included in the GRAPHICS discipline, the students acquire a substantial body of knowledge

10. Evaluation / Grading (if applicable)

Activity type	Evaluation criteria	Evaluation methods	Percentage of the final grade
10.1. Lecture	**The correct solution of applications and problems** **The correctness of the answers** **correct understanding and application of the issues covered in the course**	**Grid questions based on practical applications that solve various problems chosen by the student**	70%
10.2. Seminar	The correct solution of homework during the semester.	Testing programs in the presence of students	20% 10%
10.3. Laboratory			
10.4. Project			
10.5 Minimal performance standard **- grade 5 (five) for completing 70% of the laboratory assignments and correct answers for the exam grid to 50% of the questions.**			

Course coordinator
dr. Gabor Andrei-Marius

Seminar/laboratory/project coordinator
dr. Gabor Andrei-Marius

Head of the Department
Lector Popa Lorena

Dean
Prof.univ.dr. Sorin-Florin NĂDĂBAN

MINISTERUL EDUCAŢIEI
UNIVERSITATEA „AUREL VLAICU“ DIN ARAD
310130 Arad, B-dul Revoluției nr. 77, P.O. BOX 2/158 AR
Tel : 0040-257- 283010; fax. 0040-257- 280070
http://www.uav.ro; e-mail: rectorat@uav.ro
Operator de date cu caracter personal nr.2929

SYLLABUS

1. Study programme

1.1. Higher education institution	**„Aurel Vlaicu” University of Arad**
1.2. Faculty	**of Exact Sciences**
1.3. Department	**Department of Mathematics and Computer Science**
1.4. Field of study	**Informatics**
1.5. Study year	**2023-2024**
1.6. Study level	**Bachelor**
1.7. Study programme / Qualification	**Computer Science (in English)**
1.8. Form of education	**Full – Time study**

2. Course details

2.1. Name of the course	**GIBS6A16 Computer Science project management**
2.2. Course coordinator	**Dr Gaspar Octavian-Pastorel**
2.3. Seminar/laboratory/project coordinator	**Software developer Florea Marcela-Florina**
2.4. Study year	**3**
2.5. Semester	**2**
2.6. Evaluation type	**summative**
2.7. Course type	**optionl**

3. Estimated total time (hours per semester)

3.1. Hours per week	**3**
3.2. Lecture hours per week	**2**
3.3. Seminar/laboratory/project hours per week	**1**
3.4. Total hours per curriculum	**42**
3.5. Lecture hours per semester	**28**
3.6. Seminar/laboratory/project hours per semester	**14**
Time division [hrs]	
3.4.1. Independent study from textbooks, course support, bibliography and notes	**40**
3.4.2. Additional reading (libraries, specialized electronic platforms and field research)	**20**
3.4.3. Preparing of seminars/laboratories/projects, homework, papers, portfolios and essays	**20**
3.4.4. Tutorial coaching	**0**
3.4.5. Examinations	**3**
3.4.6. Other activities	**0**
3.7. Total individual study hours	**83**
3.8. Total hours per semester	**125**
3.9. Number of ECTS credits	**5**

4. Prerequisites (if applicable)

4.1. Curriculum related	
4.2. Competence related	

5. Conditions (if applicable)

5.1. for the lecture	**Lecture classroom with video projector.**
5.2. for the seminar	
5.3. for the laboratory	**Laboratory classroom with Internet, Visual Studio installed on laptops and video projector.**
5.4. for the project	

6. Specific educational objectives (competences to be acquired)

6.1. Professional competencies	**C1. Programming in high level programming languages;** **C2. Development and maintenance of computer applications;** **C3. Using computer tools in interdisciplinary context;** **C4. Using the theoretical bases of computers and formal models;** **C5. Database design and database management;**
6.2. Transversal competencies	**CT1. Applying the rules of organized and efficient work, of responsible attitudes towards teaching-scientific field, to value the own creative potential, while respecting the principles and norms of professional ethics.** **CT2. Efficient conduct of the activities organized in an inter-disciplinary group and developing the personal communication skills, networking and collaboration with various groups;** **CT3. Using of efficient methods and techniques for learning, informing, research and development of the capacity to value knowledge, adapting to the requirements of a dynamic society and communicating in English and in an Internationally widespread language.**

7. Course outcomes (resulting from the specific educational objectives to be acquired)

7.1. General outcomes	**Aquiring the knowledge and skills necessary to manage IT projects**
7.2. Specific outcomes	**Identifying the main elements in a project, understanding and managing the elements that define the success of a project: time, budget and purpose, determining the aspects that make Agile methodologies superior to classic IT project management methodologies.**

8. Outline (if applicable)

8.1 Lecture Outline	Teaching methods	Remarks
1. What is a project in general?. Introduction to project management.	Exposure, explanation, exemplification, debate	
2. Planning activities within IT projects	Exposure, explanation, exemplification, debate	
3. The triangle: purpose, time, budget	Exposure, explanation, exemplification, debate	
4. Difficulties in project planning and evaluation	Exposure, explanation, exemplification, debate	
5. Project initiation processes	Exposure, explanation, exemplification, debate	
6. Project planning activities	Exposure, explanation, exemplification, debate	
7. Elements of project execution	Exposure, explanation, exemplification, debate	
8. Monitoring of IT projects	Exposure, explanation, exemplification, debate	

9. Closing the projects	Exposure, explanation, exemplification, debate	
10. Human resources management	Exposure, explanation, exemplification, debate	
11. Communication	Exposure, explanation, exemplification, debate	
12. Risk management	Exposure, explanation, exemplification, debate	
13. Organizational structure	Exposure, explanation, exemplification, debate	
14. Using the Agile method of IT project management	Exposure, explanation, exemplification, debate	

8.2 Lecture References

David Farley, Modern Software Engineering: Doing What Works to Build Better Software Faster, Publisher: Addison-Wesley Professional; 1st edition (December 6, 2021)
Robert K. Wysocki, Effective Project Management: Traditional, Agile, Extreme, Hybrid 8th Edition, Publisher: Wiley; 8thedition (May 7, 2019)
A Guide to the Project Management Body of Knowledge (PMBOK(R) Guide–Sixth Edition / Agile Practice Guide BundleSixth edition, Project Management Institute; Sixth edition (September 22, 2017)
Jonathan Cook, Effective Project Management for Software Development: Fine Tuning Software Project Management forOptimum Results, December 8, 2016
Lecture notes - SUMS, 2023

8.3 Seminar Outline	Teaching methods	Remarks

8.4 Seminar References

8.5 Laboratory Outline	Teaching methods	Remarks
Developing of a software project in a team: Conception and elaboration as a project manager of the plan of a medium-sized software project with a given theme (trend analysis, design, risk analysis, project organization, configuration management, planning, etc.,	explanation, example, debate, dialogue	The software project is completed during the 14 hours of laboratory, gradually, in relation to what was taught in the course.

8.6 Laboratory Bibliography

David Farley, Modern Software Engineering: Doing What Works to Build Better Software Faster, Publisher: Addison-Wesley Professional; 1st edition (December 6, 2021)
Robert K. Wysocki, Effective Project Management: Traditional, Agile, Extreme, Hybrid 8th Edition, Publisher: Wiley; 8thedition (May 7, 2019)
A Guide to the Project Management Body of Knowledge (PMBOK(R) Guide–Sixth Edition / Agile Practice Guide BundleSixth edition, Project Management Institute; Sixth edition (September 22, 2017)
Jonathan Cook, Effective Project Management for Software Development: Fine Tuning Software Project Management forOptimum Results, December 8, 2016
Lecture notes - SUMS, 2023

8.7 Project Outline	Teaching methods	Remarks

8.8 Project Outline

9. Correlation of course outline with the expectations of the epistemic community, professional associations and representative employers within the field of the program

The content of the discipline is consistent with what is done in other university centers in the country and abroad. In order to better adapt the content of the discipline to the requirements of the labor market, meetings were held both with representatives of the business environment and with mathematics and computer science teachers from pre-university education.

10. Evaluation / Grading (if applicable)

Activity type	Evaluation criteria	Evaluation methods	Percentage of the final grade
10.1. Lecture	• **Correctness and completion of gained knowledge;** • **Logical coherence;** • **Degree of assimilation of specific terms.** **Criteria concerning attitude aspects: thoroughness, interst for individual study**	Oral evaluation Active participations at lectures	40% 5%
10.2. Seminar			
10.3. Laboratory	• **Capacity of operating with gained notions;** • **Capacity of practical application.** **Criteria concerning attitude aspects: thoroughness, interst for individual study**	Software Project Active participation at seminars	50% 5%
10.4. Project			

10.5 Minimal performance standard

Minimum performance standard: knowledge of the fundamental elements of theory and practice, solving a simple application.

Course coordinator
Conf.univ. Dr.
Octavian-Pastorel
GASPAR

Seminar/laboratory/project coordinator
Software developer Florea Marcela-Florina

Head of the Department
Lector Popa Lorena

Dean
Prof.univ.dr. Sorin-Florin NĂDĂBAN

MINISTERUL EDUCAŢIEI ȘI CERCETĂRII
UNIVERSITATEA „AUREL VLAICU“ DIN ARAD
310130 Arad, B-dul Revoluției nr. 77, P.O. BOX 2/158 AR
Tel : 0040-257- 283010; fax. 0040-257- 280070
http://www.uav.ro; e-mail: rectorat@uav.ro
Operator de date cu caracter personal nr.2929

SYLLABUS

1. Study programme

1.1. Higher education institution	**„AUREL VLAICU” UNIVERSITY OF ARAD**
1.2.Faculty	**Faculty of Exact Sciences**
1.3. Department	**Department of Mathematics and Computer Science**
1.4. Field of study	**Informatics**
1.5. Academic year	**2024-2025**
1.6. Study level	**Licence**
1.7. Study programme	**Informatics -english**
1.8. Form of education	**Full-time education**

2. Course details

2.1. Name of the course	Concurrency
2.2. Course coordinator	Conf. dr. Barna Cornel
2.3. Seminar/laboratory/project coordinator	laboratory
2.4. Study year	III
2.5. Semester	6
2.6. Evaluation type	Exam
2.7. Course type	optional

3. Estimated total time (hours per semester)

3.1. Hours per week	3
3.2. Lecture hours per week	2
3.3. Seminar/laboratory/project hours per week	1
3.4. Total hours per curriculum	42
3.5. Lecture hours per curriculum	28
3.6. Seminar/laboratory/project hours per curriculum	14
	Time division [Hours]
3.4.1. Independent study from textbooks, course support, bibliography and notes	44

3.4.2. Additional reading	20
3.4.3. Preparing of seminars/laboratories/projects, homework, papers, portfolios and essay	10
3.4.4. Tutorial coaching	5
3.4.5. Examinations	4
3.4.6. Other activities	0
3.7. Total individual study hours	83
3.8. Total hours per semester	125
3.9. Number of ECTS credits	5

4. Prerequisites (if applicable)

4.1. Curriculum related	
4.2. Competence related	

5. Conditions (if applicable)

5.1. Conditions for the lecture	Classroom with laptops and video projector
5.2. Conditions for the seminar	
5.3. Conditions for the laboratory	Laboratory with computers
5.4. Conditions for the project	

6. Specific educational objectives (competences to be acquired)

6.1. Professional competences	C1. high level programming skills
6.2. Transversal competences	CT1 The application of the rules of organized and efficient work, of responsible attitudes towards the didactic-scientific field, for the creative exploitation of one's own potential, in compliance with the principles and norms of professional ethics; CT3 The use of effective methods and techniques of learning, information, research and development of the capacities to capitalize on knowledge, to adapt to the requirements of a dynamic society and to communicate in an internationally used language.

7. Course outcomes (resulting from the specific educational objectives to be acquired)

7.1. General outcomes	Knowledge of the characteristics, components and utility of parallelism and distributed IT systems
7.2.Specific outcomes	. The motivation for using computer systems distributed on the Internet.

8. Course outline

8.1 Lecture	Teaching methods	Remarks
1. PARALLEL, CONCURRENT AND DISTRIBUTED COMPUTING PARADIGMS 2. DESIGN REQUIREMENTS OF A DISTRIBUTED COMPUTER SYSTEM 3. DISTRIBUTED COMPUTER SYSTEMS 4. INTERNET AND WWW AS DISTRIBUTED SYSTEMS 5. CLUSTER COMPUTING 6. GRID COMPUTING 7. CLOUD COMPUTING	Presentation, description, explanations, examples, dialogue	4 hours

8.2 Lecture references

1. Wen-mei W. Hwu (Author), David B. Kirk (Author), Izzat El Hajj (Author), Programming Massively Parallel Processors: A Hands-onApproach 4th Edition ,Publisher : Morgan Kaufmann; 4th edition (August 18, 2022)
2. A.D. Kshemkalyani, M. Singhal, Distributed Computing, Cambrige University Press, 2015 (pdf)
3. Ioan Dziţac, Grigor Moldovan, Sisteme distribuite: Modele informatice, Ed. Univ. Agora, 2006.
4. A. S. Tanenbaum, M. van Steen, Distributed Systems: Principles and Paradigms, Vrije Univ., Amsterdam,Olanda:http://www.cs.vu.nl/~ast/books/ds1/samples.html
5. Ioan Dziţac, Parallel and Distributed Methods for Algebraic Systems Resolution, Ed. Univ. Agora, 2006.
6. https://www.coursera.org/

8.3 Seminar	Teaching methods	Remarks

8.4 Seminarreferences

8.5 / Laboratory	Teaching methods	Remarks
1 -14. Practical aspects based on the topics discussed in the course	Computer example. Functionality testing	28 hours

8.6 Laboratory references

1. Wen-mei W. Hwu (Author), David B. Kirk (Author), Izzat El Hajj (Author), Programming Massively Parallel Processors: A Hands-onApproach 4th Edition ,Publisher : Morgan Kaufmann; 4th edition (August 18, 2022)
2. A.D. Kshemkalyani, M. Singhal, Distributed Computing, Cambrige University Press, 2015 (pdf)
3. Ioan Dziţac, Grigor Moldovan, Sisteme distribuite: Modele informatice, Ed. Univ. Agora, 2006.
4. A. S. Tanenbaum, M. van Steen, Distributed Systems: Principles and Paradigms, Vrije Univ., Amsterdam,Olanda:http://www.cs.vu.nl/~ast/books/ds1/samples.html
5. Ioan Dziţac, Parallel and Distributed Methods for Algebraic Systems Resolution, Ed. Univ. Agora, 2006.
6. https://www.coursera.org/

8.7 / Project	Teaching methods	Remarks

8.8 Project References

9. Corroboration / validation of course putline(if applicable)

The content of the discipline is consistent with what is done in other university centers in the country and abroad. In order to better adapt the content of the subject to the requirements of the labor market, meetings were held both with representatives of the business environment and with mathematics and computer science teachers from pre-university education.

10. Evaluation / Grading

Activitytype	Evaluation criteria	Evaluation methods	Percentage of the final grade
10.1. Lecture	- correctness and completeness of knowledge; -logical coherence; - degree of assimilation of the specialized language;	Written paper	50%
10.2. Seminar			
10.3. Laboratory	Knowledge and understanding; - Ability to explain and interpret; - Complete and correct solution of the requirements.	- Certified application activity / laboratory / practical works - Tests during the semester	50%
10.4. Project			
10.5 Minimal performance standard Minimum performance standard: knowledge of the fundamental elements of theory and practice, solving a simple application.			

Titular	Asistent	DIRECTOR DEPARTAMENT	DECAN
Conf.dr.Barna Cornel	Conf.dr.Barna Corne	Lect. dr. Lorena Popa	Prof. dr. Nadaben Sorin