# SYLLABUS

## 1. Program Information

| | |
|---|---|
| 1.1 Higher education institution | Technical University of Cluj-Napoca |
| 1.2 Faculty | Faculty of Automation and Computer Science |
| 1.3 Department | Department of Automation |
| 1.4 Field of study | Automation, Applied Informatics and Intelligent Systems |
| 1.5 Cycle of studies | Bachelor |
| 1.6 Study Programme/Qualification | Intelligent Automation Systems (dual, in English language) |
| 1.7 Form of education | IF – full-time education |
| 1.8 Course code | 9.00 |

## 2. Course information

| | | | | | | |
|---|---|---|---|---|---|---|
| 2.1 Course title | | | | | **Computer Architecture and Operating Systems** | |
| 2.2 Course lecturer | | | | | *Lecturer Eng. Vasile-Cosmin Marcu, Ph.D.,* *Cosmin.Marcu@aut.utcluj.ro* | |
| 2.3 Seminar / Laboratory / Project Lecturer | | | | | *Dr.ing. Sergiu Pascu (Emerson)* | |
| 2.4 Year of study | 1 | 2.5 Semester | 2 | 2.6 Type of assessment | | E |
| 2.7 Course status | Formative category *(DF, DS, DC)* | | | | | DF |
| | Optionality *(DOB, DOP, DFac)* | | | | | DOB |

## 3.Total estimated time

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 3.1 Number of hours per week | 4 | of which: | HEI | Lecture | 2 | Seminar | 0 | Laboratory | 0 | Project | 0 |
| | | | CO | | 0 | | 0 | | 2 | | 0 |
| 3.2 Number of hours per semester | 56 | of which: | HEI | Lecture | 28 | Seminar | 0 | Laboratory | 0 | Project | 0 |
| | | | CO | | 0 | | 0 | | 28 | | 0 |

| 3.3 Distribution of time allocation (hours per semester) for: | HEI | CO |
|---|---|---|
| (a) Study based on textbook, course support, bibliography, and notes | 10 | 10 |
| (b) Additional documentation in library, specialized electronic platforms, and fieldwork | 15 | 9 |
| (c) Preparation of seminars/laboratories, assignments, papers, portfolios and essays | 20 | 0 |
| (d) Tutoring | 0 | 0 |
| (e) Examinations | 2 | 3 |
| (f) Other activities: | 0 | 0 |
| 3.4 Total individual study hours (sum (3.3(a)... 3.3(f))) | 47 | 22 |
| 3.5 Total hours per semester (3.2+3.4) | 75 | 50 |
| 3.6 Number of credits per semester | 3 | 2 |

*(HEI = Higher Education Institution, CO = Company)*

## 4. Prerequisites (where applicable)

| | |
|---|---|
| 4.1 Curriculum Prerequisites | Computer Programming and Algorithm Design, Mathematics |
| 4.2 Competency Prerequisites | Analytical, logical, and systems-level reasoning. Ability to independently explore technical documentation, troubleshoot problems, and manage lab tasks. Ability to solve programming tasks. Understand and apply conditional logic. |

## 5. Conditions (where applicable)

| | |
|---|---|
| 5.1. Course Organization Conditions | Blackboard, Multimedia projector, PC |
| 5.2. Seminar / Laboratory / Project organization conditions | PC, Specific software, specific hardware |

**6. Specific Competencies Acquired**

| | |
|---|---|
| Professional Competencies | • PC06, Define technical requirements<br>• PC07, Demonstrate disciplinary expertise<br>• PC17, Operate open source software<br>• PC23, Synthesise information<br>• PC24, Think abstractly<br>• PC26, Use information technology tools |
| Transversal Competencies | • TC02, Think analitically<br>• TC03, Demonstrate responsability |

**7. Learning outcomes**

| | |
|---|---|
| Knowledge | • Describe the fundamental components and organization of modern computer architectures, including CPU, memory hierarchy, and I/O subsystems.<br>• Explain the principles of operating systems with a focus on Windows OS internals: processes, threads, memory, and file system management.<br>• Understand Windows kernel architecture, including the interaction between user mode and kernel mode components.<br>• Describe virtualization concepts, including hardware-assisted virtualization, and how Windows operates in virtualized environments.<br>• Understand key system-level security features in Windows. |
| Skills | • Analyze and monitor system performance, memory usage, and process behavior in Windows.<br>• Configure and manage virtual machines using Hyper-V.<br>• Investigate and troubleshoot Windows system events, device drivers, and boot configuration.<br>• Automate Windows system tasks using PowerShell scripts and registry modifications.<br>• Apply system design knowledge to optimize and evaluate system performance in both native and virtualized environments. |
| Responsibility and autonomy | • Independently install, configure, and secure Windows systems in both physical and virtual environments.<br>• Take initiative in analyzing system behaviors and proposing solutions for performance, compatibility, or security issues.<br>• Collaborate in a team setting to design, implement, and present a virtualization or OS-level project.<br>• Demonstrate responsible use of system-level tools, especially those with administrative privileges, maintaining system integrity and ethical standards.<br>• Critically and independently assess the impact of virtualization on security, performance, and system design decisions in real-world scenarios. |

**8. Course Objectives**

| | |
|---|---|
| 8.1 General objective of the course | This course provides a comprehensive understanding of computer architecture fundamentals and operating system |

| | |
|---|---|
| | principles, with a specialized focus on the Windows operating system and virtualization technologies. Students will learn how hardware and software interact at a low level, how Windows manages processes, memory, and I/O, and how modern virtualization platforms operate. |
| 8.2 Specific objectives | • Core computer architecture concepts<br>• System resources management<br>• Basic networks<br>• Process and thread management<br>• Memory management including virtual memory<br>• File systems, registry, and security management<br>• Set up and manage virtual machines using Hyper-V<br>• Impact of virtualization on performance, security, and system architecture. |

**9. Contents**

| 9.1 Lectures | No. of hours | Teaching methods | Obs. |
|---|---|---|---|
| Introduction and historical perspective | 2 | Presentation using blackboard and multimedia projector; Example applications | - |
| General structure of a computer | 2 | | |
| Data representation and conversion | 2 | | |
| Storage, installation and accessibility | 2 | | |
| Introduction to operating systems | 2 | | |
| Computer networks | 2 | | |
| Active Directory (users, groups, domains policies) | 2 | | |
| Virtual systems | 2 | | |
| Files, directories and processes | 2 | | |
| Software deploys using PowerShell | 2 | | |
| System services configuration. Windows Registry | 2 | | |
| System and Hardware security. Maintenance and patching | 2 | | |
| Disaster recovery | 2 | | |
| System Architecture Design | 2 | | |

Bibliography:

[1] D. A. Patterson and J. L. Hennessy, *Computer Organization and Design: The Hardware/Software Interface*, 5th ed., Amsterdam, Netherlands: Morgan Kaufmann, 2013.
[2] J. L. Hennessy and D. A. Patterson, *Computer Architecture: A Quantitative Approach*, 6th ed., Amsterdam, Netherlands: Morgan Kaufmann, 2019.
[3] A. S. Tanenbaum and T. Austin, *Structured Computer Organization*, 6th ed., Boston, MA, USA: Pearson, 2013.
[4] P. Yosifovich, M. E. Russinovich, D. A. Solomon, and A. Ionescu, *Windows Internals, Part 1: System Architecture, Processes, Threads, Memory Management, and More*, 7th ed., Redmond, WA, USA: Microsoft Press, 2021.
[5] A. Silberschatz, P. B. Galvin, and G. Gagne, *Operating System Concepts*, 10th ed., Hoboken, NJ, USA: Wiley, 2018.
[6] A. S. Tanenbaum and H. Bos, *Modern Operating Systems*, 4th ed., Upper Saddle River, NJ, USA: Pearson, 2014.
[7] J. M. Hart, *Windows System Programming*, 6th ed., Boston, MA, USA: Addison-Wesley, 2015.
[8] A. Syrewicze and R. Siddaway, *Hands-On Hyper-V 2022: Build, Maintain, and Optimize Virtual Infrastructure*, Birmingham, UK: Packt Publishing, 2022.

| 9.2 Seminar / laboratory / project | Hours HEI | Hours CO | Teaching methods | Obs. |
|---|---|---|---|---|
| Evolution of computer | - | 2 | Interactive teaching tools. Specific software for systems configuration; practical usecases. | |
| General Structure of Modern Computer Systems | - | 2 | | |
| Data representation and conversion. Floating point conversion | - | 2 | | |
| Storage Management and OS Installation | - | 2 | | |
| Operating Systems Fundamentals | - | 2 | | |
| Basic Computer Networks. IP addresses and subnets | - | 2 | | |
| User and Group Management in Active Directory: Policies and Domain Control | - | 2 | | |
| Deploying and Managing Virtual Systems | - | 2 | | |
| File Systems and Process Management in a Multi-User Environment | - | 2 | | |
| Automated Software Deployment Using PowerShell Scripts | - | 2 | | |
| System Services and Registry Configuration | - | 2 | | |
| System Security and Maintenance: Patching and Hardware Protection | - | 2 | | |
| Disaster Recovery Plan: Strategies and Simulation | - | 2 | | |
| System architecture design: Principles and Practical Models | - | 2 | | |

Bibliography

[1] J. M. Hart, *Windows System Programming*, 6th ed., Boston, MA, USA: Addison-Wesley, 2015.
[2] P. Yosifovich, M. E. Russinovich, D. A. Solomon, and A. Ionescu, *Windows Internals, Part 1: System Architecture, Processes, Threads, Memory Management, and More*, 7th ed., Redmond, WA, USA: Microsoft Press, 2021.
[3] A. Syrewicze and R. Siddaway, *Hands-On Hyper-V 2022: Build, Maintain, and Optimize Virtual Infrastructure*, Birmingham, UK: Packt Publishing, 2022.
[4] T. Lammle, *CompTIA Network+ Certification Guide: Exam N10-008*, 2nd ed., Birmingham, UK: Packt Publishing, 2022.

**10. Correlation of course content with the expectations of the epistemic community representatives, professional associations, and major employers in the field related to the program**

The course content is aligned with other technical universities' programs and provides the necessary knowledge to deal with industry and employers requirements for computer architecture and operating systems.

**11. Evaluation**

| **Activity Type** | **Evaluation criteria** | **Evaluation methods** | **Weight in final grade** |
|---|---|---|---|
| 11.1 Lecture | Conceptual understanding, applied knowledge, problems solving, analysis | Written exam | 50% |
| 11.2 Seminar/ Laboratory/Project | Practical execution skills, conceptual explanation, troubleshooting and debugging, use of tools and commands | Written tests and practical applications | 50% |
| 11.3 Minimum Performance Standard | | | |
| 5 out of 10 points | | | |

| Date of completion: | Lecturers | | Signature |
|---|---|---|---|
| 15.05.2025 | Course<br>Applications | *Lecturer Ph.D. Eng. Vasile-Cosmin Marcu* | |

| | |
|---|---|
| Date of approval by the Department of Automation Council<br>24.11.2025 | Director of the Department of Automation<br>Prof.dr.ing. Honoriu VĂLEAN |
| Date of approval by the Faculty of Automation and Computer Science Council<br>28.11.2025 | Dean<br>Prof.dr.ing. Vlad MUREȘAN |