

## SYLLABUS OPERATING SYSTEMS

### 1. Information about the Program

|  |                                      |
|--|--------------------------------------|
| 1.1 Higher education institution                   | Politehnica University of Timișoara  |
| 1.2 Faculty <sup>1</sup> / Department <sup>2</sup> | Automation and Computers/ Computers  |
| 1.3 Chair  | -                                    |
| 1.4 Domain of study                                | Computers and Information Technology |
| 1.5 Study level                                    | Bachelor                             |
| 1.6 Study programme / Qualification                | Computers / engineer                 |

### 2. Information about the Course

|                                      |  |              |   |                     |   |                 |           |
|--------------------------------------|--|--------------|---|---------------------|---|-----------------|-----------|
| 2.1 Course                           | Operating Systems                        |              |   |                     |   |                 |           |
| 2.2 Lecturer                         | dr. Dan C. Cosma                         |              |   |                     |   |                 |           |
| 2.3 Academic staff for seminars/labs | dr. Dan C. Cosma, dr. Casandra Holotescu |              |   |                     |   |                 |           |
| 2.4 Study year                       | 3  | 2.5 Semester | 1 | 2.6 Assessment type | E | 2.7 Course type | mandatory |

### 3. Total time estimated (hours/ semester of didactical activities)

|   |    |           |                   |    |                       |       |
|---|----|-----------|-------------------|----|-----------------------|-------|
| 3.1 Hours / week  | 4  | of which: | 3.2 lecture hours | 2  | 3.3 seminar/lab hours | 2     |
| 3.4 Total curriculum hours  | 56 | of which: | 3.5 lecture hours | 28 | 3.6 seminar/lab hours | 28    |
| Time distribution   |    |           |                   |    |                       | hours |
| Study using manuals, support materials, bibliography and notes                          |    |           |                   |    |                       | 28    |
| Supplementary documentation in library, speciality electronic platforms and on site     |    |           |                   |    |                       | 10    |
| Supplementary preparation for seminars/labs, homeworks, reviews, portofolios and essays |    |           |                   |    |                       | 27    |
| Tutoring activities   |    |           |                   |    |                       | 14    |
| Exams   |    |           |                   |    |                       | 2     |
| Other   |    |           |                   |    |                       |       |
| <b>3.7 Total - hours of individual study</b>  |    |           |                   |    |                       | 65    |
| <b>3.8 Total - hours per semester</b>   |    |           |                   |    |                       | 137   |
| <b>a. Credits</b>   |    |           |                   |    |                       | 5     |

### 4. Prerequisites (if appropriate)

|                  |   |
|------------------|---|
| 4.1 curriculum   | <ul style="list-style-type: none"> <li>Computer Programming</li> </ul>  |
| 4.2 competencies | <ul style="list-style-type: none"> <li>C programming, basic knowledge about computer software and hardware</li> </ul> |

### 5. Conditions (if appropriate)

|                       |  |
|-----------------------|--|
| 5.1 for lectures      | <ul style="list-style-type: none"> <li>Lecture room, laptop, projector, screen, whiteboard, course website</li> </ul>  |
| 5.2 for seminars/labs | <ul style="list-style-type: none"> <li>PCs/laptops for individual study, UNIX-based software platform, projector, screen, whiteboard, laboratory web site</li> </ul> |

<sup>1</sup> Se înscrie numele facultății care gestionează programul de studiu căruia îi aparține disciplina;

<sup>2</sup> Se înscrie numele departamentului căruia i-a fost încredințată susținerea disciplinei și de care aparține titularul cursului;

## 6. Specific competencies acquired

|  |   |
|--|---|
| Professional competencies <sup>3</sup> | <ul style="list-style-type: none"> <li>• Operarea cu fundamente științifice, ingineresti și ale informaticii</li> <li>• Proiectarea componentelor hardware, software și de comunicații</li> <li>• Soluționarea problemelor folosind instrumentele științei și ingineriei calculatoarelor</li> <li>• Îmbunătățirea performanțelor sistemelor hardware, software și de comunicații</li> <li>• Proiectarea, gestionarea ciclului de viață, integrarea și integritatea sistemelor hardware, software și de comunicații</li> <li>• Proiectarea sistemelor inteligente</li> </ul> |
| Transversal competencies               | <ul style="list-style-type: none"> <li>• Comportarea onorabilă, responsabilă, etică, în spiritul legii pentru a asigura rezolvarea problemei</li> <li>• Identificarea, descrierea și derularea proceselor din managementul proiectelor, cu preluarea diferitelor roluri în echipă și descrierea clară și concisă, verbal și în scris, în limba română și într-o limbă de circulație internațională, a rezultatelor din domeniul de activitate</li> </ul>  |

## 7. Objectives of the course (issued from the list of the competencies acquired)

|                         |   |
|-------------------------|---|
| 7.1 Aim                 | <ul style="list-style-type: none"> <li>• Understand the concepts related to operating systems, the OS role, functionality and structure, work with its services as a professional, understand and learn system programming</li> </ul>   |
| 7.2 Specific objectives | <ul style="list-style-type: none"> <li>• A.1 Understand the relevance and importance of operating systems and system programming in the context of modern computing</li> <li>• A.2 Understand the important concepts related to operating systems, and their impact on software systems</li> <li>• A.3 Learn to program using system services</li> <li>• A.4 Exploiting the operating system in a professional and efficient manner</li> <li>• A.5 Solve specific problems using software engineering techniques in the context of system services</li> <li>• A.6 Automate tasks through programming and scripting</li> <li>• A.7 Understand and assimilate the patterns, techniques and strategies specific to system programming</li> </ul> |

## 8. Content

| 8.1 Lecture     | Hours | Instruction methods |
|-----------------|-------|---------------------|
| 1. Introduction | 4     |                     |

<sup>3</sup> Aspectul competențelor profesionale va fi tratat cf. Metodologiei OMECTS 5703/18.12.2011. Se vor prelua competențele care sunt precizate în Registrul Național al Calificărilor din Învățământul Superior RNCIS ([http://www.rncis.ro/portal/page?\\_pageid=117,70218&\\_dad=portal&\\_schema=PORTAL](http://www.rncis.ro/portal/page?_pageid=117,70218&_dad=portal&_schema=PORTAL)) pentru domeniul de studiu de la pct. 1.4 și programul de studii de la pct. 1.6 din această fișă.

|  |    |  |
|--|----|--|
| 1.1 Definitions, context, history<br>1.2 Overview of operating systems<br>1.3 Types of Operating Systems<br>1.4 UNIX, Linux, OS X, Windows<br>1.5 Responsibilities and functionalities<br>1.6 Execution environments. Virtualization<br>1.7 Users and rights. Authentication and authorization. User spaces<br>1.8 Memory management                 |    | <p>The course involves a combination of heuristic and expository teaching strategies (problematization, explanation, conversation, discovery, experimentation, etc.) along with algorithmic procedures (generalizing patterns and techniques for problem solving, methods of structuring and implementing software applications, etc.).</p> <p>The course follows a pragmatic approach, and places the student at the core of the activities. It consists of a set of interactive lectures where feedback is sought as often as possible. The approach is mainly inductive, the detailed examples and technology-specific information working together to build an environment that allows the student to understand the main concepts and gain the necessary abilities.</p> |
| 2. Using Operating Systems<br><br>2.1 Users, administrators and power users<br>2.2 User interfaces<br>2.3 Introduction to system administration<br>2.4 Programming interfaces. System and library calls  | 2  |  |
| 3. The command-line interface<br><br>3.1 Purpose and benefits<br>3.2 The UNIX command line<br>3.3 The Windows command line<br>3.4 Shell scripting. Regular expressions   | 4  |  |
| 4. File systems<br>4.1 Definitions. Characteristics<br>4.2 Storage media. Types of file systems<br>4.3 Hierarchical organization. File types.<br>4.4 Device abstraction<br>4.5 Permissions and ownership<br>4.6 File system layout. File attributes<br>4.7 Programming with files. Descriptors: UNIX, Windows. Standard input and output. Attributes | 4  |  |
| 5. Processes<br><br>5.1 Concepts<br>5.2 Processes. Process states. Scheduling<br>5.3 Process groups. Process attributes. Inheritance<br>5.4 Programming with processes: UNIX, Windows  | 2  |  |
| 6. Inter-process communication<br><br>6.1 IPC<br>6.2 Signals<br>6.3 Communication through files. File locks<br>6.4 Pipes<br>6.5 System V IPC   | 6  |  |
| 7. Threads<br><br>7.1 Concepts<br>7.2 Multithreading<br>7.3 Programming with threads: UNIX, Windows  | 2  |  |
| 8. Advanced issues<br><br>8.1 OS Design goals<br>8.2 OS Architectures<br>8.3 Asynchronous I/O<br>8.4 Terminal I/O<br>8.5. Elements of system administration  | 4  |  |
| Total:   | 28 |  |

|   |       |   |
|---|-------|---|
|   |       |   |
|   |       |   |
|   |       |   |
| References  |       |   |
| <p>1. W.R.Stevens, S.A.Rago, <i>Advanced Programming in the UNIX Environment, Third Edition</i>; Addison Wesley, 2013<br/> 2. W. Stallings, <i>Operating Systems: Internals and Design Principles, 7<sup>th</sup> edition</i>, Prentice Hall, 2011<br/> 3. A. Robbins: <i>UNIX in a Nutshell, Fourth Edition</i>; O'Reilly, 2005<br/> 4. A. S. Tannenbaum: <i>Modern Operating Systems, 2<sup>nd</sup> Edition</i>, Prentice Hall, 2001<br/> 5. K. Haviland, D. Gray, B. Salama: <i>UNIX System Programming, 2<sup>nd</sup> edition</i>, Addison-Wesley, 1998</p>   |       |   |
| 8.2 Seminar/lab   | Hours | Instruction methods   |
| 1. Introduction   | 2     | <p>During the lab classes, students solve, at the computer, a problem specific to the current chapter. The process is supervised by the teacher, through presentations, discussions, and answering questions. At the end of the class, students present their work.</p> |
| 2. Shell scripting. Filters. Regular expressions  | 6     |   |
| 3. Programming with files and directories   | 6     |   |
| 4. Programming with processes. Inter-process communication  | 11    |   |
| 5. Threads  | 3     |   |
|   |       |   |
| References  |       |   |
| <p>1. W.R.Stevens, S.A.Rago, <i>Advanced Programming in the UNIX Environment, Third Edition</i>; Addison Wesley, 2013<br/> 2. W. Stallings, <i>Operating Systems: Internals and Design Principles, 7<sup>th</sup> edition</i>, Prentice Hall, 2011<br/> 3. A. Robbins: <i>UNIX in a Nutshell, Fourth Edition</i>; O'Reilly, 2005<br/> 4. A. S. Tannenbaum: <i>Modern Operating Systems, 2<sup>nd</sup> Edition</i>, Prentice Hall, 2001<br/> 5. K. Haviland, D. Gray, B. Salama: <i>UNIX System Programming, 2<sup>nd</sup> edition</i>, Addison-Wesley, 1998<br/> 6. Online lab resources (complete lab guide), posted at labs.cs.upt.ro</p> |       |   |

**9. Correlation between the course content and the requirements of the specialists in the field and the expectations of the main employers**

|  |
|--|
| <ul style="list-style-type: none"> <li>• Learning system programming and how to use an OS professionally are essential for any software specialist. The diversity of the OS market, and their rapid evolution makes abilities such understanding the OS-related concepts, knowing the mechanisms involved and understanding the provided servicessoftware essential for designers and developers.</li> <li>• Most modern software applications use directly or indirectly the srvcies of the OS. A presumptive lack of knowledge in the field of Operating Systems may generate serious problems for developing and maintaining software systems. Thus, knowing the concepts and techniques involved in interfacing the applications wth the OS are essential abilities in the software development market.</li> </ul> |
|--|

**10. Assessment**

| Activity type | 10.1 Assessment criteria  | 10.2 Assessment methods   | 10.3 Weight in final mark |
|---------------|---|---|---------------------------|
| 10.4 Lecture  | The written exam captures the degree of understanding the concepts, the abilities gained through the semester, and the ability of solving problems and making connections throughout the field. | Written exam comprising of questions that require short to medium-sized answers. Questions are focused on clear subjects, and assess both the understanding of the concepts, and the ability of correlating the studied techniques, methods, etc. | 0.65                      |

|  |   |  |      |
|--|---|--|------|
|  |   |  |      |
| 10.5 Seminar /labs   | Students solve specific problems throughout the semester, by implementing programs on the computer that make use of OS services. The evaluation is done through pre-scheduled tests, where students must write programs which are evaluated by the teacher. During tests, students must answer questions related to their work and the involved concepts. | Students receive grades for each test, The final grade is a weighted average of the marks for the tests. | 0.35 |
| 10.6 Minimal performance standards (minimal specific knowledge required for passing the exam, the means to assess mastering the specific knowledge)  |   |  |      |
| <ul style="list-style-type: none"> <li>• understanding the main concepts taught,</li> <li>• having the ability to write programs using system services,</li> <li>• knowing to solve problems using the techniques and tools introduced by the course and lab.</li> </ul> |   |  |      |

**11. International compatibility**

- |   |
|---|
| <ul style="list-style-type: none"> <li>• University of Cambridge, UK, Operating Systems, <a href="http://www.cl.cam.ac.uk/teaching/1112/OpSystems/">http://www.cl.cam.ac.uk/teaching/1112/OpSystems/</a></li> <li>• University of Washington , Introduction to Operating Systems, (<a href="http://www.cs.washington.edu/education/abet_syllabi/CSE451.html">http://www.cs.washington.edu/education/abet_syllabi/CSE451.html</a> )</li> <li>• Virginia Tech, Operating Systems, (<a href="http://courses.cs.vt.edu/%7Ecs3204/">http://courses.cs.vt.edu/%7Ecs3204/</a> )</li> </ul> |
|---|

Date

Signature of the course instructor

Signatures of the academic staff for seminars/labs

.....

.....

Date of approval in the Department

Signature of the Department Director

.....