

## SYLLABUS

### 1. Information regarding the programme

1.1 Higher education institution	Babes-Bolyai University
1.2 Faculty	Mathematics and Informatics
1.3 Department	Informatics
1.4 Field of study	Informatics
1.5 Study cycle	Licence
1.6 Study programme / Qualification	Informatics - english

### 2. Information regarding the discipline

2.1 Name of the discipline (en) (ro)	Operating systems Sisteme de operare						
2.2 Course coordinator	Assoc. prof. Rares Boian						
2.3 Seminar coordinator	Assoc. prof. Rares Boian						
2.4. Year of study	1	2.5 Semester	2	2.6. Type of evaluation	E	2.7 Type of discipline	Mandatory
2.8 Code of the discipline	MLE5007						

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

3.1 Hours per week	5	Of which: 3.2 course	2	3.3 seminar/laboratory	3
3.4 Total hours in the curriculum	70	Of which: 3.5 course	28	3.6 seminar/laboratory	42
Time allotment:					hours
Learning using manual, course support, bibliography, course notes					20
Additional documentation (in libraries, on electronic platforms, field documentation)					10
Preparation for seminars/labs, homework, papers, portfolios and essays					10
Tutorship					5
Evaluations					10
Other activities: .....					
3.7 Total individual study hours	55				
3.8 Total hours per semester	125				
3.9 Number of ECTS credits	5				

### 4. Prerequisites (if necessary)

4.1. curriculum	.
4.2. competencies	.

### 5. Conditions (if necessary)

5.1. for the course	· The requirements posted here <a href="http://www.cs.ubbcluj.ro/~rares/course/os/">http://www.cs.ubbcluj.ro/~rares/course/os/</a>
5.2. for the seminar /lab activities	· Lab rooms with Windows and UNIX operating system access · The requirements posted here <a href="http://www.cs.ubbcluj.ro/~rares/course/os/">http://www.cs.ubbcluj.ro/~rares/course/os/</a>

## 6. Specific competencies acquired

<b>Professional competencies</b>	<ul style="list-style-type: none"> <li>· Define notions, concepts, theories and models of basic operating systems.</li> <li>· Critical analysis and use of the principles, methods and techniques work for quantitative and qualitative evaluation of the processes within an operating system and communication mechanisms between its processes</li> <li>· Apply basic concepts and theories in the field of computer architecture, programming methods and operating systems project development professional</li> <li>· Ability to solve problems for low-level interface on OS kernels</li> </ul>
<b>Transversal competencies</b>	<ul style="list-style-type: none"> <li>· Execution of the tasks required under specified requirements and the deadlines imposed, with the rules of professional ethics and moral conduct</li> <li>· Information and permanent documentation in its field</li> <li>· Seeking to improve business results by engaging in professional activities</li> </ul>

## 7. Objectives of the discipline (outcome of the acquired competencies)

7.1 General objective of the discipline	<ul style="list-style-type: none"> <li>· Learning the main entities and concepts that operates in operating systems: processes and files.</li> <li>· The base station processes the binding of files and communication between processes.</li> <li>· Learning programming bases specific operating systems: type scripting language programming (sh, bash, PowerShell) and use system functions in standard C language.</li> <li>· Presentation of the case studies, the enitator, concepts and APIs operating processes provided by the Unix family of operating systems (Solaris, Linux, BSD, etc)</li> </ul>
7.2 Specific objective of the discipline	<ul style="list-style-type: none"> <li>· Unix operating system: introduction.</li> <li>· Programming Shell.</li> <li>· Windows Introduction: bat file commands</li> <li>· Unix file system: tree structure and connections.</li> <li>· Unix Operating System I / O, processes signals.</li> <li>· Communication between Unix processes.</li> <li>· Installation and configuration of operating systems</li> </ul>

## 8. Content

8.1 Course	Teaching methods	Remarks
Week. 1 Unix: Introduction. Unix commands and arguments. Regular Expression, generic specification of files. Filters and text editors.	<ul style="list-style-type: none"> <li>· Interactive exposure</li> <li>· Explanation</li> <li>· Conversation</li> <li>· Didactical demonstration</li> </ul>	
Week. 2 Shell Programming. Sh Processors. Variables, control structures (if, for, while, do, case). Embedded commands.	<ul style="list-style-type: none"> <li>· Interactive exposure</li> <li>· Explanation</li> <li>· Conversation</li> </ul>	

Remarcable shell variables.	<ul style="list-style-type: none"> <li>· Didactical demonstration</li> </ul>	
Week. 3. Windows: introduction. Commands and arguments. Files and paths; Access rights Command bat files	<ul style="list-style-type: none"> <li>· Interactive exposure</li> <li>· Explanation</li> <li>· Conversation</li> <li>· Didactical demonstration</li> </ul>	
Week. 4 OS Unix: processes. Unix processes; structure, API (fork, wait, exec, exit, system, popen).	<ul style="list-style-type: none"> <li>· Interactive exposure</li> <li>· Explanation</li> <li>· Conversation</li> <li>· Didactical demonstration</li> </ul>	
Week. 5 POSIX Threads Concepts. API: create, exit, join. Mutex variables.	<ul style="list-style-type: none"> <li>· Interactive exposure</li> <li>· Explanation</li> <li>· Conversation</li> <li>· Didactical demonstration</li> </ul>	
Week. 6 Unix File System; I/O operations. Hard and symbolic links. Mounting. File access rights open, close, read, write, lseek, file lock.	<ul style="list-style-type: none"> <li>· Interactive exposure</li> <li>· Explanation</li> <li>· Conversation</li> <li>· Didactical demonstration</li> </ul>	
Week. 7 General Theory of Operating Systems Classifications. Functions Architectures.	<ul style="list-style-type: none"> <li>· Interactive exposure</li> <li>· Explanation</li> <li>· Conversation</li> <li>· Didactical demonstration</li> </ul>	
Week. 8 Processes. Concepts Concurrence. Semaphores. Critical sections and race conditions Deadlock. Processes scheduling	<ul style="list-style-type: none"> <li>· Interactive exposure</li> <li>· Explanation</li> <li>· Conversation</li> <li>· Didactical demonstration</li> </ul>	
Week. 9 Memory management Architecture Alocatiopns: partitioned, paging, segmentation. Swapping Memory scheduling	<ul style="list-style-type: none"> <li>· Interactive exposure</li> <li>· Explanation</li> <li>· Conversation</li> <li>· Didactical demonstration</li> </ul>	
Week. 10 Phisical I/O I/O chanel Zone tampon. Disk access scheduling	<ul style="list-style-type: none"> <li>· Interactive exposure</li> <li>· Explanation</li> <li>· Conversation</li> <li>· Didactical demonstration</li> </ul>	

Week. 11 File Systems Concepts Low-level implementations. Directories Jurnalization; copy-on-write Example: FAT, EXT3, NTFS	<ul style="list-style-type: none"> <li>· Interactive exposure</li> <li>· Explanation</li> <li>· Conversation</li> <li>· Didactical demonstration</li> </ul>	
Week. 12 Operating systems booting	<ul style="list-style-type: none"> <li>· Interactive exposure</li> <li>· Explanation</li> <li>· Conversation</li> <li>· Didactical demonstration</li> </ul>	
Week. 13 Linux kernel	<ul style="list-style-type: none"> <li>· Interactive exposure</li> <li>· Explanation</li> <li>· Conversation</li> <li>· Didactical demonstration</li> </ul>	
Week. 14 Windows kernel	<ul style="list-style-type: none"> <li>· Interactive exposure</li> <li>· Explanation</li> <li>· Conversation</li> <li>· Didactical demonstration</li> </ul>	

#### Bibliography

1. ALBING C., VOSSEN J.P., NEWHAM C. bash Cookbook. O'Reilly, 2007
  2. BOIAN F, VANCEA A. BOIAN R. BUFNEA D., STERCA A., COBARZAN C., COJOCAR D. Sisteme de operare Ed. Risoprint, 2006.
  3. BOIAN F.M. De la aritmetica la calculatoare. Ed. Presa Universitara Clujeana, Cluj, 1996.
  4. BOIAN F.M. FERDEAN C.M., BOIAN R.F., DRAGOS R.C. Programare concurentă pe platforme Unix, Windows, Java. Ed. Albastră, grupul Microinformatica, Cluj, 2002.
  5. BOIAN F.M. Servicii web; modele, platforme, aplicații. Ed. Albastră - grupul Microinformatica, Cluj, 2012
  6. LUTZ M. Learning Python. O'Reilly, 2009.
  7. RAYMOND E.S. The Art of Unix Programming. Prentice Hall, 2003.
  8. STALLINGS W. Operating Systems: Internal and Design Principles. 6th edition, Prentice Hall, 2009.
  9. TANENBAUM A.S. Modern Operating Systems. 3rd edition, Prentice Hall, 2009
  10. Ubuntu - The Complete Reference. Richard Petersen, MCGraw-Hill, 2009
- Windows 7 User Guide. Microsoft, 2009

8.2 Seminar / laboratory	Teaching methods	Remarks
Unix: commands and text editors	<ul style="list-style-type: none"> <li>· Interactive exposure</li> <li>· Explanation</li> <li>· Conversation</li> <li>· Didactical demonstration</li> </ul>	
sed, grep, awk	<ul style="list-style-type: none"> <li>· Interactive exposure</li> <li>· Explanation</li> <li>· Conversation</li> </ul>	

	<ul style="list-style-type: none"> <li>· Didactical demonstration</li> </ul>	
Shell program	<ul style="list-style-type: none"> <li>· Interactive exposure</li> <li>· Explanation</li> <li>· Conversation</li> <li>· Didactical demonstration</li> </ul>	
C program under Unix using gcc	<ul style="list-style-type: none"> <li>· Interactive exposure</li> <li>· Explanation</li> <li>· Conversation</li> <li>· Didactical demonstration</li> </ul>	
Windows bat	<ul style="list-style-type: none"> <li>· Interactive exposure</li> <li>· Explanation</li> <li>· Conversation</li> <li>· Didactical demonstration</li> </ul>	
Unix processes	<ul style="list-style-type: none"> <li>· Interactive exposure</li> <li>· Explanation</li> <li>· Conversation</li> <li>· Didactical demonstration</li> </ul>	
Unix threads	<ul style="list-style-type: none"> <li>· Interactive exposure</li> <li>· Explanation</li> <li>· Conversation</li> <li>· Didactical demonstration</li> </ul>	
Unix; thread + mutex	<ul style="list-style-type: none"> <li>· Interactive exposure</li> <li>· Explanation</li> <li>· Conversation</li> <li>· Didactical demonstration</li> </ul>	
Windows processes	<ul style="list-style-type: none"> <li>· Interactive exposure</li> <li>· Explanation</li> <li>· Conversation</li> <li>· Didactical demonstration</li> </ul>	
Windows threads	<ul style="list-style-type: none"> <li>· Interactive exposure</li> <li>· Explanation</li> <li>· Conversation</li> <li>· Didactical demonstration</li> </ul>	

Closing lab activities	<ul style="list-style-type: none"> <li>· Interactive exposure</li> <li>· Explanation</li> <li>· Conversation</li> <li>· Didactical demonstration</li> </ul>	
Practical exam	<ul style="list-style-type: none"> <li>· Interactive exposure</li> <li>· Explanation</li> <li>· Conversation</li> <li>· Didactical demonstration</li> </ul>	

#### Bibliography

1. ALBING C., VOSSEN J.P., NEWHAM C. bash Cookbook. O'Reilly, 2007
2. BOIAN F, VANCEA A. BOIAN R. BUFNEA D., STERCA A., COBARZAN C., COJOCAR D. Sisteme de operare Ed. Risoprint, 2006.
3. BOIAN F.M. De la aritmetica la calculatoare. Ed. Presa Universitara Clujeana, Cluj, 1996.
4. BOIAN F.M. FERDEAN C.M., BOIAN R.F., DRAGOS R.C. Programare concurentă pe platforme Unix, Windows, Java. Ed. Albastră, grupul Microinformatica, Cluj, 2002.
5. RAYMOND E.S. The Art of Unix Programming. Prentice Hall, 2003.
6. Ubuntu - The Complete Reference. Richard Petersen, MCGraw-Hill, 2009

#### **9. Corroborating the content of the discipline with the expectations of the epistemic community, professional associations and representative employers within the field of the program**

- By learning the theoretical and methodological concepts and addressing the practical aspects of the Operating Systems course, students acquire a body of knowledge consistent, consistent with partial competencies required for possible occupations provided in Grid 1 - RNCIS
- The course complies with IEEE and ACM Curricula Recommendations for Computer Science studies.
- The course curriculum exists in universities and faculties in Romania
- The course content is very well appreciated by software companies whose employees and graduates of this course

#### **10. Evaluation**

Type of activity	10.1 Evaluation criteria	10.2 Evaluation methods	10.3 Share in the grade (%)
10.4 Course	The level of knowledge and understanding of the course subjects	Written exam	30%
	Problem solving		
10.5 Seminar/lab activities	Ability to solve practical problems, specific to the course subjects, on the computer in a given amount of time	Practical exams in the middle of the semester and the last two weeks	25%
	Lab activity	Tests and projects	35%

<b>10.6 Minimum performance standards</b>
---

Ø A minimum of 5 in the final grade
-------------------------------------

Date

10.5.2016

Signature of course coordinator

Assoc. prof. Rares Boian

Signature of seminar coordinator

Assoc. prof. Rares Boian

Date of approval

.....

Signature of the head of department

Prof. Dr. Anca Andreica