



## COURSE DESCRIPTION

### 1. Program Information

1.1 University	University “Alexandru Ioan Cuza” of Iași
1.2 Faculty	Faculty of Computer Science
1.3 Department	Department of Computer Science
1.4 Study Domain	Computer Science
1.5 Study Cycle	Bachelor's degree
1.6 Study Program / Qualification	Computer Science/ B.Sc. degree in Computer Science

### 2. Course Information

2.1 Course Name	Special chapters on operating systems						
2.2 Course Teacher	Lecturer Cristian Vidrașcu, PhD						
2.3 Seminary Teacher	Lecturer Cristian Vidrașcu, PhD						
2.4 Study Year	3	2.5 Semester	2	2.6 Evaluation	M	2.7 Course Status*	OP

\* OB – Compulsory / OP – Optional

### 3. Total estimated hours (hours per semester and didactic activities)

3.1 Hours per week	2	in which: 3.2 course	2	3.3 seminary/laboratory	2
3.4 Hours in curriculum	56	in which: 3.5 course	28	3.6 seminary/laboratory	28
Time Distribution					hours
Manual study, Course support, Bibliography, and others					30
Supplementary Documentation in library, in electronic forums, and on the field					20
Seminaries/laboratories preparation, homeworks, reports, portfolios and essays					30
Tutoring					-
Evaluation					4
Other activities					10
3.7 Total hours individual study					80
3.8 Total hours per semester					150
3.9 Credits					5

### 4. Preconditions (if necessary)

4.1 Of Curriculum	Computer architecture, Operating systems, Computer networks, Programming (C/C++ language)
4.2 Of Skills	Specific skills acquired in 4.1

### 5. Conditions (if necessary)

5.1 For Course Operation	--
5.2 For Seminary/Laboratory Operation	--



## 6. Specific Skills Acquired

<b>Professional Skills</b>	<p><b>C1.</b> The description of concepts and basic models used in the design and implementation of operating systems.</p> <p><b>C2.</b> The identification and explanation of the basic components of Windows kernel.</p> <p><b>C3.</b> The use of mathematical and computer science models and tools for solving some specific operating systems problems.</p> <p><b>C4.</b> The development of software components for applications which use the Windows kernel's services.</p>
<b>Transversal Skills</b>	<p><b>CT1.</b> The efficient conduct of work activities, showing responsible attitudes toward the scientific and teaching domain, for the creative exploitation of their potential, with respect for the principles and rules of professional ethics.</p>

## 7. Course Objectives (from the grid of specific skills acquired)

<b>7.1 General Objectives</b>	<ol style="list-style-type: none"><li>1. Deepening the knowledge about operating systems, on techniques used for the design and implementation of them.</li><li>2. Presenting advanced topics about Windows kernel modules. Concepts are presented based on curriculum developed by Microsoft Windows Academic Program, structured according to ACM/IEEE Operating System Body of Knowledge.</li><li>3. Acquiring abilities for programmatic use of the services offered by Windows kernel.</li></ol>
<b>7.2 Specific Objectives</b>	<p>On successful completion of this subject, students will be able to:</p> <ul style="list-style-type: none"><li>• Explain the basic concepts related to the operating systems.</li><li>• Describe the Windows operating system architecture, its main components and services.</li><li>• Analyse the algorithms that are used by Windows kernel for resource management.</li><li>• Utilize application programming interface provided by Windows.</li><li>• Design software applications that use the services provided by Windows kernel.</li></ul>

**8. General Description**

8.1	Course	Teaching Methods	Observations (hours and bibliographic references)
1.	The evolution of operating systems. Windows family - concepts and tools	Exposure (lecture with illustrations and applications, explanation) using Powerpoint presentations and demonstrations at the blackboard.	2
2.	Structure of Windows kernel	Idem	2
3.	Windows subsystems. Windows API	Idem	2
4.	The basic mechanisms of Windows kernel. Concurrency (critical sections, semaphores)	Idem	2
5.	Interrupt handling and synchronization mechanisms in Windows.	Idem	2
6.	Processes and threads in Windows. Internal structures used for their management	Idem	2
7.	Thread scheduling in Windows	Idem	2
8.	Recap	-	-
9.	Fundamentals of memory management in Windows	Idem	2
10.	Virtual address translation. Managing physical memory	Idem	2
11.	Principles of I/O system. The structure and operation of the I/O system in Windows	Idem	2
12.	System security in Windows - components and concepts, security descriptors	Idem	2
13.	Fundamentals of filesystems	Idem	2
14.	The NTFS filesystem	Idem	2

**Bibliography****Main references:**

- Mark Russinovich and David Solomon, *Windows Internals, 6th edition - Part 1&2*, Microsoft Press, 2012.

**Supplementary references:**

- Windows Academic Program: <http://www.microsoft.com/resources/sharedsource/windowsacademic/default.mspx>

8.2	Seminary / Laboratory	Teaching methods	Observations (hours and bibliographic references)
1.	Windows API Overview	Exposure, exercises, individual work	2
2.	API for files and registry (Part I)	Idem	2
3.	API for files and registry (Part II)	Idem	2
4.	API for managing processes and threads (Part I)	Idem	2
5.	API for managing processes and threads (Part II)	Idem	2
6.	Synchronization mechanisms (Part I)	Idem	2
7.	Synchronization mechanisms (Part II)	Idem	2
8.	Recap	-	-
9.	API for network communications (Part I)	Idem	2
10.	API for network communications (Part II)	Idem	2
11.	Client-server applications (Part I)	Idem	2
12.	Client-server applications (Part II)	Idem	2
13.	Writing a kernel-mode driver (Part I)	Idem	-
14.	Writing a kernel-mode driver (Part II)	Idem	2

**Bibliography**

- Mark Russinovich and David Solomon, *Windows Internals, 6th edition - Part 1&2*, Microsoft Press, 2012.

- MSDN: <http://msdn.microsoft.com>

**9. Course content synchronization with the expectations of the community representatives, professional associations and employers from the program domain**

Course content provides the necessary skills to work on the Windows platform, the dominant platform used as a target for the development of commercial applications by IT companies.

**10. Evaluation**

<b>Activity Type</b>	<b>10.1 Evaluation criteria</b>	<b>10.2 Evaluation methods</b>	<b>10.3 The weight of each evaluation form (%)</b>
<b>10.4</b> Course	Ability to apply theoretical elements to solve practical problems.	written test	33,3%
<b>10.5</b> Seminary/ Laboratory	The ability to use WinAPI calls and to develop programs for Windows platform.	six lab projects	66,7%
<b>10.6</b> Minimal performance standards			
<ul style="list-style-type: none"><li>- Operating system architecture: minimal knowledge about the Windows kernel components and provided services, and about the algorithms underlying their operation</li><li>- Use of the operating system: minimal knowledge of the WinAPI interface offered by Windows to access its kernel services</li><li>- Implementation and documentation of program units in the C/C++ programming language for programmatic access to the services provided by a Windows system</li> <li>- To promote you have to obtaine at least a minimum threshold (which will be communicated to the first class and posted on the course webpage) of the maximum score that can be achieved, both at the written test and at the laboratory tests.</li><li>- Final grades are computed according to the ECTS criteria.</li></ul>			

Date  
19.05.2014

Course Teacher  
Lecturer Cristian Vidrașcu, PhD

Seminary/Laboratory Teacher  
Lecturer Cristian Vidrașcu, PhD

Department Date of Approval

Director of the Department  
Professor Dorel Lucanu, PhD