

**SYLLABUS / FIȘA DISCIPLINEI**
**1. Information on the study programme / Date despre programul de studii**

1.1. Institution / Instituția de învățământ superior	Universitatea de Vest din Timișoara
1.2. Faculty / Facultatea	Matematică și Informatică
1.3. Department / Departamentul	Computer Science (Informatică)
1.4. Study program field	Computer Science (Informatică)
1.5. Study cycle/ Ciclul de studii	Bachelor / licență
1.6. Study programme / Programul de studii / calificarea*	Computer Science / Informatică în limba engleză / <i>Computer network administration / Administrator de rețea de calculatoare - 252301; Analyst / Analist - 251201; Research assistant in computer science / Asistent de cercetare în informatica - 214918; Teacher in secondary schools / Profesor în învățământul gimnazial - 233002; Programmer / Programator - 251202; Software systems designers / Proiectant sisteme informatice - 251101</i>

**2. Information on the course / Date despre disciplină**

2.1. Title of the course / Denumirea disciplinei	Computer Architecture						
2.2. Teacher in charge of the course / Titularul activităților de curs	Cristian Cira						
2.3. Teacher in charge of the seminar / Titularul activităților de seminar	Cristian Cira						
2.4. Study year / Anul de studii	<b>2</b>	2.5. Semester / Semestrul	<b>1</b>	2.6. Examination type / Tipul de evaluare: Exam/Colloquim	<b>E</b>	2.7. Course type / Regimul disciplinei: M(andatory)/ E(lective)/ F(acultative)	<b>D</b>

**3. Estimated study time (number of hours per semester) /Timpul total estimat (ore pe semestru al activităților didactice)**

3.1. Attendance hours per week / Număr de ore pe săptămână	4	out of which din care: 3.2 lecture/ curs	2	3.3. seminar/laborator	2
3.4. Attendance hours per semester / Total ore din planul de învățământ	56	out of which: 3.5 lecture / curs	28	3.6. seminar/laborator	28
<b>Distribution of the allocated amount of time / Distribuția fondului de timp*</b>					<b>hours / ore</b>
Individual study /Studiu după manual, suport de curs, bibliografie și notițe					14

Supplementary documentation at library or using electronic repositories / Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate	28
Preparing for laboratories, homework, reports etc. /Pregătire seminarii/laboratoare, teme, referate, portofolii și eseuri	28
Exams / Examinări	4
Tutoring / Tutorat	2
3.7. Total number of hours of individual study / Total ore studiu individual	76
3.8. Total number of hours per semester / Total ore pe semestru	132
3.9. Number of credits (ECTS) / Număr de credite	5

#### 4. Prerequisites (if it is the case) / Precondiții (acolo unde e cazul)

4.1. curriculum / de curriculum	No prerequisites
4.2. skills / de competențe	Proficiency in English, Analytical mindset, Ability to decompose complex problems into sub-problems, basic understanding of Mathematics, Logic, Physics and/or Chemistry.

#### 5. Requirements (if it is the case) / Condiții (acolo unde e cazul)

5.1. for the lecture / de desfășurare a cursului	Good internet connection, e-uvt.ro students account, Google Classroom and Google Meet
5.2. for the seminar, laboratory / de desfășurare a seminarului/laboratorului	Good internet connection, e-uvt.ro students account, Google Classroom and Google Meet

#### 6. Acquired skills / Competențe specifice acumulate

Professional skills / Competențe profesionale	<ul style="list-style-type: none"> <li>● Fundamental building blocks of computing</li> <li>● Introduction to von Neumann Architectures</li> <li>● History of Computer Science</li> </ul>
Transversal skills / Competențe transversale	<ul style="list-style-type: none"> <li>● Ability to analyze and evaluate the design or performance of compute systems.</li> <li>● Ability to exploit hardware specification to optimize software for performance</li> </ul>

#### 7. Objectives of the course / Obiectivele disciplinei (reieșind din grila competențelor specifice acumulate)

7.1. General objective / Obiectivul general al disciplinei	This course aims to introduce freshman to the challenges, solutions, and alternatives offered by various architecture and design decisions that shaped the history of computing: <ul style="list-style-type: none"> <li>● understand the basics,</li> <li>● understand the principles (of design),</li> </ul>
--	---

	<ul style="list-style-type: none"> <li>understand the precedents (in computer architecture).</li> </ul>
7.2. Specific objectives / Obiectivele specifice	<p><i>Knowledge objective (KO):</i> (1) understand of von Neuman architecture components and their interactions; (2) familiarity to various abstraction levels from low-level hardware to application-level software; (3) basic understanding Information technology future hardware trends; (4) learn how a modern computer works underneath, from the bottom up.</p> <p><i>Ability objectives (AO):</i> (1) evaluate tradeoffs of different designs and ideas; (2) understand processor micro-architecture and pipelining; (3) evaluate cache and virtual memory organizations; (4) learn to systematically debug increasingly complex systems; (5) evaluate memory models (7) use parallel computers.</p> <p><i>Skill objectives (SK):</i> (1) evaluate and/or assemble a desktop computer within a performance budget (2) learn how a modern computer works underneath, from the bottom up</p>

### 8. Content / Conținuturi\*

8.1. Lecture / Curs	Teaching strategies / Metode de predare	Remarks, details / Observații
<b>Recommended bibliography / Bibliografie</b>		
8.2. Seminar, lab / Seminar, laborator	Teaching / learning strategies / Metode de predare / învățare	Remarks, details / Observații
C1. Introduction & Logistics (2h)	Lecture, discussions, active student participation	Online or face-to-face
C2. History of computing and Computer Architecture (2h)	Same as above	Online or face-to-face
C3. Combinatorial Logic (2h)	Same as above	Online or face-to-face
C4. Sequential Logic (2h)	Same as above	Online or face-to-face
C5. Timing and Verification (2h)	Same as above	Online or face-to-face

C6. Von Neumann Model, ISA, LC-3 and MIPS (2h)	Same as above	Online or face-to-face
C7. Instruction Set Architecture (2h)	Same as above	Online or face-to-face
C8. Microarchitecture (2h)	Same as above	Online or face-to-face
C9. Pipelining & Out-of-order Execution (2h)	Same as above	Online or face-to-face
C10. Cache (2h)	Same as above	Online or face-to-face
C11. Memory Hierarchy (2h)	Same as above	Online or face-to-face
C12. Virtual Memory (2h)	Same as above	Online or face-to-face
C13. Multiprocessor Cache (2h)	Same as above	Online or face-to-face
C14. Alternatives to Von Neumann Model (2h)	Same as above	Online or face-to-face

**Recommended bibliography / Bibliografie**

1. Digital Design and Computer Architecture (ETH Zürich, Spring 2020 - [YouTube](#))
2. Computerphile (YouTube channel)
3. Crash Course: Computer Science (YouTube channel)
4. David A. Patterson, John L. Hennessy -- Computer Organization and Design: The Hardware Software Interface [RISC-V Edition]
5. Andrew Tanenbaum , Todd Austin -- Structured Computer Organization [6th Edition]

<b>8.2 Seminar, laboratory</b>	<b>Teaching methods</b>	<b>Remarks, details</b>
L1-13 (26h) Complementary discussions to the lectures. Homework assessments. Open questions and comments from students. Work on mid-term and final projects	Active participation, discussions, self-conducted practical work	Online
L14 (2h) Final knowledge evaluation		

**9. Correlations between the content of the course and the requirements of the IT field / Coroborarea conținuturilor disciplinei cu așteptările reprezentanților comunității epistemice, asociațiilor profesionale și angajatorilor reprezentativi din domeniul aferent programului**

This is a basic course in computer science programs all around the world. Questions from this course appear both in the graduation exam, as well as in hiring interviews.

--

**10. Evaluation / Evaluare\***

Activity / Tip de activitate	10.1. Evaluation criteria / Criterii de evaluare**	10.2. Evaluation methods / Metode de evaluare***	10.3. Weight in the averaged mark
10.4. Lecture/Cur s	<ul style="list-style-type: none"> <li>● Familiarity with the topics introduced during the semester</li> <li>● Understanding of common design tradeoffs that impact performance</li> <li>● Understanding the role that each level of abstraction plays in computer architecture</li> </ul>	At least 2 individual semester projects (5 min videos) At least 1 multiple choices test	50%  10%
10.5. Seminar/lab	<ul style="list-style-type: none"> <li>● Exercises and questions on the week's topic</li> </ul>	At least 10 individually graded homeworks as Google classroom submission	40%
10.6. Minimal knowledge for passing / Standard minim de performanță			
Minimal knowledge for passing this subject: <ul style="list-style-type: none"> <li>● Ability to converse for 5 minutes with any IT professional about [some] performance tradeoffs in computer architecture design</li> </ul> <p>The final grade is computed as a weighted average of grades obtained for components described in 10.4 and 10.5.</p>			

 Date/ Data completării  
**08.09.2020**

 Signature (lecture) /  
 Semnătura titularului de curs

 Signature (seminar)  
 Sem. titularului de seminar

 Signature (director of the department)  
 Semnătura directorului de departament