FIȘA DISCIPLINEI

1. Date despre program

Facultatea	Facultatea de Inginerie Electrică și Știința Calculatoarelor
Departamentul	Calculatoare, electronică şi automatică
Domeniul de studii	Automatică, informatică aplicată şi sisteme inteligente
Ciclul de studii	Licență, dual
Programul de studii	Automatică şi informatică aplicată

2. Date despre disciplină

Denumirea disciplinei	**SISTEME DE OPERARE**					
Anul de studiu	III	Semestrul	5	Tipul de evaluare	C	
Regimul disciplinei	Categoria formativă a disciplinei DF - fundamentală, DS - de specializare, DC – complementară					DS
	Categoria de opționalitate a disciplinei: DOB – obligatorie, DOP – opțională, DFA - facultativă					DOP

3. Timpul total estimat (ore alocate activităților didactice)

I a) Număr de ore pe săptămână	4	Curs	2	Seminar		Laborator/ Lucrări practice IIS		Proiect IIS		Practică IIS	
						Laborator/ Lucrări practice IM	2	Proiect IM		Practică IM	
I b) Totalul de ore pe semestru din planul de învățământ	56	Curs	28	Seminar		Laborator	28	Proiect		Practică	

(*IIS – instituție de învățământ superior; IM – învățare prin muncă*)

Distribuția fondului de timp pe semestru	Ore IIS	Ore IM
II.Studiu individual	20	45
III. Examinări	2	2
IV. Alte activități (precizați):		

Total ore studiu individual (II + III)	Ore IIS	22	Ore IM	47
Total ore pe semestru (Ib+II+III+IV)	Ore IIS	50	Ore IM	75
Numărul de credite	Credite IIS	2	Credite IM	3

4. Competențe specifice acumulate

Competențe transversale	
Competențe profesionale	CP2. Proiectează componente de automatizare CP5. Proiectează o rețea de calculatoare

5. Rezultatele învățării

Cunoștințe	Aptitudini	Responsabilitate și autonomie
Studentul/absolventul descrie evoluția și principiile fundamentale ale sistemelor de operare, caracterizează principalele tipuri de arhitecturi (monolitică, microkernel, hibridă) și explică relația hardware–software în executarea programelor. Studentul/absolventul identifică și clasifică procesele și firele de execuție, descrie stările și tranzițiile acestora și explică principalele politici de planificare (FCFS, SJF, RR, multinivel).	Studentul/absolventul aplică bune practici de programare concurentă privind claritatea, siguranța și testabilitatea codului și colaborează cu membrii echipei pentru analiza și îmbunătățirea soluțiilor dezvoltate. Studentul/absolventul justifică deciziile tehnice privind utilizarea resurselor de procesare, memorie și intrare/ieșire, luând în considerare criterii de eficiență, siguranță și utilizare responsabilă a resurselor.	Studentul/absolventul manifestă autonomie și inițiativă în actualizarea cunoștințelor din domeniul sistemelor de operare și în alegerea responsabilă a soluțiilor tehnice adecvate.

Studentul/absolventul analizează condițiile necesare apariției interblocării (excludere mutuală, deținere și așteptare, imposibilitatea preempției și așteptare circulară) și propune strategii de prevenire, evitare sau recuperare. Studentul/absolventul aplică mecanisme de sincronizare (mutex-uri, semafoare, monitoare) pentru a rezolva probleme clasice de concurență. Studentul/absolventul compară și evaluează mecanisme și politici de gestiune a memoriei (paginare, segmentare, paginare la cerere și politici de înlocuire), pe baza criteriilor de performanță. Studentul/absolventul explică și reprezintă schematic fluxurile operațiilor de intrare/ieșire (buffering, caching, spooling) și evaluează comportamentul dispozitivelor și al driverelor în raport cu încărcarea sistemului.	Studentul/absolventul valorifică feedbackul primit și îmbunătățește în mod iterativ soluțiile dezvoltate. Studentul/absolventul utilizează instrumente specifice sistemelor Linux și Windows pentru monitorizarea și gestionarea proceselor, memoriei și operațiilor de intrare/ieșire (de exemplu, ps, top, iostat, strace, Task Manager). Studentul/absolventul dezvoltă și testează aplicații concurente bazate pe procese și fire de execuție, utilizând mecanisme adecvate de sincronizare. Studentul/absolventul identifică, analizează și depanează scenarii de interblocare și condiții de cursă (race conditions), modifică parametrii relevanți și evaluează efectele acestora asupra comportamentului sistemului.	

6. Obiectivele disciplinei (reieșind din grila competențelor specifice acumulate)

Obiectivul general al disciplinei	● **Obiectivul principal al disciplinei** este de a oferi studenților o înțelegere aprofundată a arhitecturii și funcționalității sistemelor de operare, prin corelarea aspectelor teoretice cu aplicații practice. Disciplina urmărește familiarizarea cu evoluția și principiile fundamentale ale sistemelor de operare, evidențierea interdependenței dintre arhitectura hardware și mecanismele software, precum și aprofundarea conceptelor de procese, fire de execuție, interblocări, comunicare și sincronizare. Totodată, sunt vizate consolidarea cunoștințelor privind managementul memoriei, al dispozitivelor de intrare/ieșire și al resurselor de stocare, înțelegerea tehnicilor de gestiune a discurilor și sistemelor de fișiere, precum și aplicarea acestora în studii de caz pe sisteme de operare moderne (Linux și Windows). Prin activitățile practice, studenții își vor dezvolta competențe analitice și aplicative necesare pentru a proiecta și implementa soluții informatice eficiente în domeniul sistemelor de operare.

7. Conținutul predării și învățării

Curs	Nr. ore	Metode de predare	Observații
1. Introducere	2	expunerea, prelegerea-dezbaterea	
2. O privire de ansamblu asupra sistemelor de operare	2	expunerea, prelegerea-dezbaterea	
3. Controlul și descrierea proceselor	2	expunerea, prelegerea-dezbaterea	
4. Fire de execuție	3	expunerea, prelegerea-dezbaterea	
5. Concurența: Excluderea mutuală și sincronizarea	3	expunerea, prelegerea-dezbaterea	
6. Concurența: Deadlock (blocajele) și starvation (înfometarea)	3	expunerea, prelegerea-dezbaterea	
7. Gestiunea memoriei	3	expunerea, prelegerea-dezbaterea	
8. Memoria virtuală	3	expunerea, prelegerea-dezbaterea	
9. Planificarea în sistemele uniprocesor	2	expunerea, prelegerea-dezbaterea	
10. Planificarea în sistemele multiprocesor	3	expunerea, prelegerea-dezbaterea	
11. Gestiunea operațiilor de intrare/ieșire (I/O) și planificarea accesului la disc	2	expunerea, prelegerea-dezbaterea	

Bibliografie minimală recomandată

- W. Stallings, *Operating Systems: Internals and Design Principles*, 10th ed. Harlow, UK: Pearson, 2024.
- A. S. Tanenbaum și H. Bos, *Modern Operating Systems*, 5th ed. Boston, MA, USA: Pearson, 2022.
- R. H. Arpaci-Dusseau și A. C. Arpaci-Dusseau, *Operating Systems: Three Easy Pieces*, ver. 1.10, Nov. 2023. Arpaci-Dusseau Books. [Online]. Disponibil: https://pages.cs.wisc.edu/~remzi/OSTEP

- R. Cox, F. Kaashoek, și R. Morris, *xv6: a simple, Unix-like teaching operating system*, RISC-V book, rev. 5, Sept. 2, 2025. [Online]. Disponibil: https://pdos.csail.mit.edu/6.828/2025/xv6/book-riscv-rev5.pdf
- S. Boyd-Wickizer, R. Morris, și F. Kaashoek, *The Multicore Operating System: Principles and Practice*, Cambridge, MA, USA: MIT Press, 2023.

Aplicații IM (Laborator)	Nr. ore	Metode de predare	Observații
1. Protecția muncii. Introducere în terminologia Linux. Structura fișierelor și directoarelor în Linux.	2	Lucrări practice, experiment	Activitățile practice aferente disciplinei Sisteme de operare se desfășoară pe stații de lucru pe care este instalată nativ distribuția Linux Ubuntu. Pentru activitățile de laborator care implică dezvoltarea, compilarea și testarea programelor se utilizează Visual Studio Code (VS Code), împreună cu suita de instrumente GNU pentru dezvoltare software disponibilă în mediul Linux.
2. Interpretoare de comenzi. Comenzi shell	2	Lucrări practice, experiment	
3. Gestiunea proceselor în Linux. Apelurile de sistem fork() și execv()	2	Lucrări practice, experiment	
4. Comunicația între procese folosind pipe-uri anonime	2	Lucrări practice, experiment	
5. Comunicația între procese folosind memorie partajată	2	Lucrări practice, experiment	
6. Sincronizarea proceselor folosind semafoare	2	Lucrări practice, experiment	
7. Comunicația între procese și sincronizarea proceselor folosind mesaje	2	Lucrări practice, experiment	
8. Evaluare (test practic)	2	Lucrări practice, experiment	
9. Problema producător-consumator	2	Lucrări practice, experiment	
10. Fire de execuție (threads)	2	Lucrări practice, experiment	
11. Fire de execuție în C++	2	Lucrări practice, experiment	
12. Comunicația între procese folosind socket-uri orientate pe conexiune (socket-uri TCP)	2	Lucrări practice, experiment	
13. Examinarea activității firelor de execuție în WINDOWS	2	Lucrări practice, experiment	
14. Evaluare (test practic)	2	Lucrări practice, experiment	

Bibliografie minimală recomandată

- Ioan Ungurean, „Sisteme de operare. Îndrumar de laborator”, ON-LINE, Editura Universității “Ștefan cel Mare” Suceava, 2022, ISBN: 978-973-666-722-0. Disponibil: http://www.eed.usv.ro/~ioanu/SO2021/
- R. H. Arpaci-Dusseau și A. C. Arpaci-Dusseau, *Operating Systems: Three Easy Pieces*, ver. 1.10, Nov. 2023. Arpaci-Dusseau Books. [Online]. Disponibil: https://pages.cs.wisc.edu/~remzi/OSTEP/
- K. N. Billimoria, *Linux Kernel Programming: A Comprehensive and Practical Guide to Kernel Internals, Writing Modules, and Kernel Synchronization*, 2nd ed. Birmingham, UK: Packt, Feb. 2024.
- J.-B. Persson, *Linux System Programming Techniques: Become a Proficient Linux System Programmer Using Expert Recipes and Techniques*. Birmingham, UK: Packt, 2021.
- R. Grimm, *Concurrency with Modern C++*. Leanpub, actualizat 20 Ian. 2024. [Online]. Disponibil: https://leanpub.com/concurrencywithmodernc

13. **Evaluare**

Tip activitate	Criterii de evaluare	Metode de evaluare	Pondere din nota finală
Evaluare finală	Nivelul de însușire și înțelegere a conceptelor, principiilor și mecanismelor fundamentale ale sistemelor de operare; capacitatea de a identifica, explica, analiza și corela mecanismele de gestionare a proceselor și firelor de execuție, sincronizare, planificare, memorie, operații de intrare/ieșire și sisteme de fișiere; capacitatea de a argumenta alegerea unor soluții tehnice adecvate unor scenarii date.	Evaluare mixtă – test practic pe calculator (60%) și test-grilă online administrat prin platforma Moodle 3 (40%).	50%
Evaluare pe parcurs	Capacitatea de a utiliza mediul Linux și instrumentele specifice pentru gestionarea proceselor și resurselor; capacitatea de a implementa, testa și depana aplicații concurente care utilizează procese, fire de execuție și mecanisme de comunicare și sincronizare; interpretarea rezultatelor obținute și justificarea soluțiilor adoptate; desfășurarea autonomă și responsabilă a activităților practice; valorificarea	Evaluare continuă (prin metode orale și probe practice) + 2 teste practice	50%

	feedbackului primit și îmbunătățirea iterativă a soluțiilor; manifestarea inițiativei în identificarea și aplicarea unor soluții tehnice adecvate.		

Fișa disciplinei include, dacă este cazul, elemente adaptate persoanelor cu dizabilități, în funcție de tipul și gradul acestora.

Data completării	Grad didactic, nume, prenume, semnătura titularului de curs	Grad didactic, nume, prenume, semnătura titularului de aplicație
21.09.2026	Conf. dr. ing. Ioan UNGUREAN	as. dr. ing. Mihail TERENTI

Data avizării	Grad didactic, nume, prenume, semnătura responsabilului de program
22.09.2026	Ș.l dr. ing. Corneliu BUZDUGA

Data avizării în departament	Grad didactic, nume, prenume, semnătura directorului de departament
24.09.2026	Conf. dr. ing. Eugen COCA

Data aprobării în consiliul facultății	Grad didactic, nume, prenume, semnătura decanului
25.09.2026	Prof. dr. ing. Laurențiu-Dan MILICI