

FIȘA DISCIPLINEI

1. Date despre program

1.1 Instituția de învățământ superior	Universitatea Tehnică din Cluj-Napoca
1.2 Facultatea	Automatică și Calculatoare
1.3 Departamentul	Calculatoare
1.4 Domeniul de studii	Calculatoare și Tehnologia Informației
1.5 Ciclu de studii	Licență
1.6 Programul de studii / Calificarea	Calculatoare română / Inginer
1.7 Forma de învățământ	IF – învățământ cu frecvență

2. Date despre disciplină

2.1 Denumirea disciplinei	Proiectarea Sistemelor de Operare				Codul disciplinei	47.1
2.2 Titularul de curs	Conf. dr. ing. Coleșa Adrian - adrian.colesa@cs.utcluj.ro					
2.3 Titularul activităților de seminar / laborator / proiect	Conf. dr. ing. Coleșa Adrian - adrian.colesa@cs.utcluj.ro As. drd. ing. Császár István - Istvan.Csaszar@cs.utcluj.ro					
2.4 Anul de studiu	IV	2.5 Semestrul	7	2.6 Tipul de evaluare	E	
2.7 Regimul disciplinei	Categorii formative: <i>DF – fundamentală, DS - de specialitate, DC - complementară</i>					DS
	Opționalitate: <i>DOB - obligatorie, DOP - opțională, DFA - facultativă</i>					DOP

3. Timpul total estimat

3.1 Număr de ore pe săptămână	5	din care:	Curs	2	Seminar	-	Laborator	2	Proiect	1
3.2 Număr de ore pe semestru	70	din care:	Curs	28	Seminar	-	Laborator	28	Proiect	14
3.3 Distribuția fondului de timp (ore pe semestru) pentru:										
(a) Studiul după manual, suport de curs, bibliografie și notițe										35
(b) Documentare suplimentară în bibliotecă, pe platforme electronice de specialitate și pe teren										-
(c) Pregătire seminarii / laboratoare, teme, referate, portofolii și eseuri										41
(d) Tutoriat										2
(e) Examinări										2
(f) Alte activități:										-
3.4 Total ore studiu individual (suma (3.3(a))...3.3(f)))					80					
3.5 Total ore pe semestru (3.2+3.4)					150					
3.6 Numărul de credite					6					

4. Precondiții (acolo unde este cazul)

4.1 de curriculum	Programarea calculatoarelor, Sisteme de operare
4.2 de competențe	Scrierea de programe în C ce folosesc servicii ale unui sistem de operare

5. Condiții (acolo unde este cazul)

5.1. de desfășurare a cursului	Tabla, proiector
5.2. de desfășurare a seminarului / laboratorului / proiectului	Tabla, Calculatoare atât la laborator cât și la proiect

6. Competențele specifice acumulate

6.1 Competențe profesionale	C5 - Proiectarea, gestionarea ciclului de viață, integrarea și integritatea sistemelor hardware, software și de comunicații C5.1 - Precizarea criteriilor relevante privind ciclul de viață, calitatea, securitatea și interacțiunea sistemului de calcul cu mediul și cu operatorul uman C5.2 - Utilizarea unor cunoștințe interdisciplinare pentru adaptarea
-----------------------------	--

	sistemului informatic în raport cu cerințele domeniului de aplicații • C5.3 - Utilizarea unor principii și metode de bază pentru asigurarea securității, siguranței și usurinței în exploatare a sistemelor de calcul • C5.4 - Utilizarea adecvată a standardelor de calitate, siguranță și securitate în prelucrarea informațiilor • C5.5 - Realizarea unui proiect incluzând identificarea și analiza problemei, proiectarea, dezvoltarea și demonstrând o înțelegere a nevoii de calitate.
6.2 Competențe transversale	N/A

7. Rezultatele așteptate ale învățării

Cunoștințe	Describe, identifică, sumarizează concepte, modele și metode avansate privind sisteme și infrastructuri informatice complexe, sisteme distribuite și paralele, rețele și protocoale de comunicații, sisteme inteligente, precum și principiile managementului și metodologiei proiectelor informatice.
Aptitudini	Aplică și proiectează soluții informatice integrate, incluzând baze de date, rețele, sisteme software și hardware, sisteme distribuite/paralele, mecanisme de securitate și inteligența artificială, utilizând metode de analiză, proiectare, implementare și documentare.
Responsabilitate și autonomie:	Își asumă responsabilitatea pentru calitatea, securitatea și corectitudinea soluțiilor dezvoltate și manifestă autonomie în organizarea activităților de proiectare și învățare, colaborând eficient în echipe și adaptându-se la cerințe și tehnologii noi.

8. Obiectivele disciplinei

8.1 Obiectivul general al disciplinei	Înțelegerea funcționalității unui sistem de operare per ansamblu și a componentelor sale, precum gestionarul de procese, gestionarul de memorie, sistemul de fișiere.
8.2 Obiectivele specifice	1. Cunoașterea diferitelor niveluri și componente ale unui sistem de operare, rolul și funcționalitatea fiecăreia și relația dintre ele. 2. Cunoașterea tehnicilor și alternativelor de proiectare și implementare a diferitelor componente ale unui SO, a avantajelor și dezavantajelor lor. 3. Capacitatea de proiectare a diferitelor componente ale unui SO și a relațiilor dintre ele: planificator, mecanisme de sincronizare, procese, thread-uri, paginare, memorie virtuală, sistem de fișiere.

9. Conținuturi

9.1 Curs	Nr.ore	Metode de predare	Observații
Structura generală a unui SO. Posibilități de structurare a unui SO (monolit, pe niveluri, micro-kernel, mașină virtuală, exokernel), componentele sale, funcționalitatea și rolul lor, interconexiunea dintre ele.	2	Prezentare pe videoproiector. Detalieri, exemplificări și explicații adiționale pe tablă.	
Planificarea thread-urilor: studiul diferitelor politici și algoritmi de planificare: FCFS, SJF, bazat pe priorități, multi-level feedback queue.	2	Întrebări adresate studenților.	
Planificarea thread-urilor în diferite sisteme de operare: Solaris, Windows, Linux.	2	Teste de 10 minute, la sfârșitul orei, din subiectul prezentat	
Mecanisme de sincronizare. Principii generale de proiectare și implementare a lacetelor, semafoarelor și variabilelor conditionale. Mecanisme hardware utilizate pentru implementarea mecanismelor de sincronizare.	2	și identificarea împreună a unei soluții corecte.	

Mecanisme de sincronizare în Linux și Windows. Modul de implementare a mecanismelor de sincronizare din Linux și Windows.	2		
Interblocarea thread-urilor. Metode de evitare, prevenire și detecție a interblocării.	2		
Suport pentru aplicații utilizator (1). Aspecte generale de proiectare, apeluri sistem posibile. Tehnici și metode de proiectare și implementare a proceselor și thread-urilor.	2		
Suport pentru aplicații utilizator (2). Gestiunea fișierelor deschise. Exemplificare pe sistemul de operare Linux: nivelul Virtual File System (VFS).	2		
Gestiunea memoriei (1) . Principii generale de proiectare. Alternative de proiectare și implementare a diferitelor tehnici de gestionare a memoriei, precum paginare și segmentare.	2		
Gestiunea memoriei (2) . Proiectarea și implementarea memoriei virtuale. Algoritmi de înlocuire a paginilor.	2		
Gestiunea memoriei (3) . Studii de caz. Tehnicile de gestionare a memoriei în Linux și Windows.	2		
Sistemul de fișiere. Aspecte generale de proiectare. Alternative de proiectare și implementare a fișierelor și directoarelor, gestiunea spațiului liber. Avantaje și dezavantaje.	2		
Sistemul de fișiere în Linux și Windows. Proiectarea și implementarea sistemelor Ext2 și NTFS.	2		
Aspecte de securitate în sistemele de operare. Recapitularea subiectelor prezentate și prezentarea lor într-o vedere de ansamblu a unui SO.	2		
Bibliografie (<i>bibliografia minimală a disciplinei conținând cel puțin o lucrare bibliografică de referință a disciplinei, care există la dispoziția studenților într-un număr de exemplare corespunzător</i>) 1. Silberschatz, G. Gagne, P. B. Galvin, Operating Systems Concepts, 9th edition, Wiley, 2012, ISBN 978-1118063330 2. A. Tanenbaum, A. Woodhull. Operating Systems Design and Implementation. 3rd edition, Prentice Hall, 2006, ISBN: 0131429388 3. Daniel Pierre Bovet, Understanding Linux Kernel, O'Reilly & Associates, 2001, ISBN 0-596-00002-2 4. Resurse pe pagina moodle a cursului la adresa https://moodle.cs.utcluj.ro			
9.2.1 Aplicații (laborator)	Nr.ore	Metode de predare	Observații
Introducere în sistemul de operare (SO) folosit: instalare, compilare, rulare, testare	2	Scurte prezentări ale subiectului și principalelor puncte cheie	
Mecanisme și tehnici de depanare a unui SO	2		
Sistemul de thread-uri	2		
Mecanisme de sincronizare	2	Răspunsuri la întrebări	
Planificarea thread-urilor	2		
Suport pentru aplicațiile utilizator (1): mecanismul de apel sistem, implementarea unor apeluri sistem simple	2	Elaborarea unui plan de lucru ce cuprinde o parte de tutorial și o parte de probleme propuse.	
Suport pentru aplicațiile utilizator (2): structura executabilelor și a spațiului de memorie virtuală	2		
Suport pentru aplicațiile utilizator (3): implementarea suportului pentru procese cu mai multe thread-uri	2		
Memorie virtuală (1): înțelegerea mecanismelor generale, încărcare la cerere (lazy-loading) a conținutului proceselor.	2	Acordarea de bonusuri studenților ce rezolvă corect problemele propuse	
Memorie virtuală (2): suport pentru fișiere mapate în memorie și încărcate la cerere	2		
Memorie virtuală (3): mecanismul de swapping și algoritmi simpli de înlocuire a paginilor	2		
Sistemul de fișiere (1): înțelegerea mecanismelor generale, implementarea fișierelor extensibile	2		

Sistemul de fișiere (2): implementarea subdirectoarelor	2		
Evaluarea cunoștințelor de laborator	2		
9.2 Aplicații - proiect	Nr.ore	Metode de predare	Observații
Prezentare SO folosit, tematica și modul de lucru	1	Discuții cu studenții pe marginea temelor predate	
Evaluare și discutare design modul threads	1		
Evaluare și discutare implementare modul threads	1		
Evaluare și discutare design modul userprog	1		
Evaluare și discutare implementare modul userprog	1		
Evaluare și discutare design modul virtual-memory	1		
Evaluare și discutare implementare modul virtual-memory	1		
Bibliografie (bibliografia minimală pentru aplicații conținând cel puțin o lucrare bibliografică de referință a disciplinei care există la dispoziția studenților într-un număr de exemplare corespunzător)			
Manual al SO folosit, disponibil pe pagina moodle a cursului la adresa https://moodle.cs.utcluj.ro			

10. Coroborarea conținuturilor disciplinei cu așteptările reprezentanților comunității epistemice, asociațiilor profesionale și angajatorilor reprezentativi din domeniul aferent programului

Cursul prezintă tehnici de gestionare a resurselor hardware și software de către sistemul de operare, dar acestea sunt valabile în cazul oricăruor resurse ce trebuie gestionate similar la un alt nivel software. În plus, cursul oferă cunoștințe detaliate asupra sistemelor de operare moderne, permițând celor care îl urmează să dezvolte aplicații mai performante. În acest sens, cursul răspunde unor necesități de ordin general în domeniul IT. Cursul răspunde în mod direct cerințelor companiilor din domeniul IT care accesează în mod direct serviciile unui SO sau dezvoltă drivere sau alte componente similare. Astfel de companii sunt de exemplu cele de securitate și detecție a aplicațiilor malițioase.

11. Evaluare

Tip activitate	Criterii de evaluare	Metode de evaluare	Pondere din nota finală
Curs	Prezența și activitatea minimă la orele de curs Corectitudinea răspunsurilor la subiectele de examen Cunoașterea alternativelor de proiectare a diferitelor componente ale unui SO Capacitatea de aplicare a cunoștințelor pe cazuri concrete	Adresarea de întrebări, discutarea unor probleme și teste scurte la orele de curs Examen de tip grilă pe platforma Moodle și examinare orală, bazată pe discutarea unor probleme concrete de proiectare a unor componente ale unui SO.	50%
Seminar	-	-	-
Laborator	Prezența și activitatea minimă la orele de laborator Corectitudinea și funcționarea codului sursă aferent problemelor rezolvate Cunoașterea principalelor structuri de date și mecanisme ale sistemului de operare utilizat	Rezolvarea pe calculator a unor probleme din subiectele studiate.	15%
Proiect	Coerența și corectitudinea designului modulelor de sistem de operare dezvoltate Corectitudinea și funcționarea codului aferent modulelor de sistem de operare Capacitatea de a proiecta și implementa soluții de extindere sau îmbunătățire a unor componente ale sistemului de operare utilizat	Prezentarea documentelor de proiectare a diferitelor componente ale SO dezvoltat și a implementării realizate; implicarea și participarea în cadrul echipei de lucru.	25%

Standard minim de performanță

Criterii. Studenții trebuie să fie capabili să descrie caracteristicile interne și funcționalitatea principalelor concepte ale sistemelor de operare (SO), inclusiv: mecanisme de blocare (locks), planificare pe bază de prioritate și algoritmul round-robin (RR), apeluri de sistem (system calls), paginare și memorie virtuală.

Studenții trebuie să fie capabili să scrie cod C funcțional care implementează componente esențiale ale unui sistem de operare, cum ar fi: planificarea firelor de execuție (thread scheduling), apeluri de sistem, managementul memoriei virtuale. **Codul livrat trebuie să treacă cu succes cel puțin un test** din setul de teste furnizat.

Cerințe de prezență**Laborator**

- Minimum 12 prezențe la laboratoare pentru a putea susține examenul final în sesiunea normală (prima)
- Minimum 10 prezențe pentru a putea susține examenul în oricare dintre sesiunile de reexaminare
- Mai puțin de 10 prezențe → **ineligibil** pentru orice reexaminare de laborator în anul universitar curent

Proiect

- Minimum 6 prezențe la activitățile de proiect pentru a putea susține examenul final în sesiunea regulată
- Minimum 5 prezențe pentru a putea susține examenul în orice sesiune de reexaminare
- Mai puțin de 5 prezențe → **ineligibil** pentru orice reexaminare de laborator în anul universitar curent

Curs

- Minimum 9 prezențe la curs pentru a putea susține examenul în sesiunea regulată
- Minimum 7 prezențe pentru a putea susține examenul în sesiuni de reexaminare
- Mai puțin de 7 prezențe → **ineligibil** pentru orice reexaminare la curs în anul universitar curent

Politica de notare**Laborator**

- Minimum nota **5** la examenul final de laborator

Proiect

- Minimum media **5** la temele de proiect
- Se permite o singură predare a temei de proiect la fiecare termen limită, fie în timpul semestrului, fie într-o sesiune de reexaminare

Examenul final

- Examenul final poate fi susținut **doar după promovarea activităților de laborator și proiect**
- Minimum nota **5** la examenul final
- Minimum media finală **5** pentru promovarea cursului

Data completării	Titulari	Titlu, prenume / NUME	Semnătura
30.04.2026	Curs	Conf.dr.ing. Adrian COLEȘA	
	Aplicații	Conf.dr.ing. Adrian COLEȘA	
		As. drd. ing. Császár István	

Data avizării în Consiliul Departamentului Calculatoare	Director Departament, Prof.dr.ing. Rodica Potolea
Data aprobării în Consiliul Facultății de Automatică și Calculatoare	Decan, Prof.dr.ing. Vlad Mureșan