

SISTEME DE OPERARE
1. Date despre unitatea de curs/modul

Facultatea	Calculatoare, Informatică și Microelectronică				
Catedra/departamentul	Ingineria Software și Automatică				
Ciclul de studii	Studii superioare de licență, ciclul I				
Programul de studiu	526.3 Automatică și Informatică				
Anul de studiu	Semestrul	Tip de evaluare	Categoria formativă	Categoria de opționalitate	Credite ECTS
II (învățământ cu frecvență la zi)	3	E	S – unitate de curs de specialitate	O - unitate de curs obligatorie	4

2. Timpul total estimat

Total ore în planul de învățământ	Din care				
	Ore auditoriale		Lucrul individual		
	Curs	Laborator	Teză de an	Studiul materialului teoretic	Pregătire aplicații
120	30	30	0	30	30

3. Precondiții de acces la unitatea de curs/modul

Conform planului de învățământ	Pentru a atinge obiectivele cursului studenții trebuie să posede abilități dezvoltate în cursurile Matematice speciale, Programări, Structuri de date și algoritmi, Metode și modele de calcul, conform planului de învățământ.
Conform competențelor	Suplimentar, studenții vor poseda abilități de lucru folosind metode și instrumente matematice, tehnice și de programare, pentru înțelegerea și elaborarea componentelor unui sistem de operare, conform competențelor dezvoltate în cursurile susnumite.

4. Condiții de desfășurare a procesului educațional pentru

Curs	Pentru prezentarea materialului teoretic în sala de curs este nevoie de tablă, cretă, proiector și calculator. Nu vor fi tolerate întârzierile, precum și convorbirile telefonice în timpul cursului.
Laborator	Studenții vor perfecta rapoarte conform condițiilor impuse de indicațiile metodice. Termenul maximal de predare a lucrării de laborator – 2 săptămâni după finalizarea acesteia.

5. Competențe specifice acumulate

Competențe profesionale	CPL 1. Utilizarea de cunoștințe de matematică, fizică, tehnica măsurării, grafică inginerescă, mecanică, electrică și electronică, în ingineria sistemelor. <ul style="list-style-type: none"> • Utilizarea în comunicarea profesională a conceptelor, teoriilor și metodelor științelor fundamentale folosite în sistemele de operare. • Explicarea temelor de rezolvat și argumentarea soluțiilor din ingineria sistemelor de operare, prin utilizarea tehnicilor, conceptelor și principiilor din matematică, fizică, grafică inginerescă, inginerie electrică, electronică. • Rezolvarea problemelor uzuale din domeniul sistemelor de operare prin identificarea de tehnici, principii, metode adecvate și prin aplicarea matematicii. • Aprecierea potențialului, avantajelor și dezavantajelor unor metode și procedee din domeniul sistemelor de operare, a nivelului de documentare științifică al proiectelor și al consistenței aplicațiilor folosind tehnici matematice și alte metode științifice. • Elaborarea de proiecte în domeniul sistemelor de operare, selectând și aplicând metode matematice și alte metode științifice specific domeniului.
-------------------------	---

	<p>CPL 2. Operarea cu concepte fundamentale din știința calculatoarelor, tehnologia informației și comunicațiilor</p> <ul style="list-style-type: none"> • Descrierea funcționării și a structurii sistemelor de calcul, rețelelor de comunicații și aplicațiilor acestora în ingineria sistemelor folosind cunoștințe referitoare la limbaje, medii și tehnologii de programare, ingineria programării și instrumente specifice (algoritmi, scheme, modele, protocoale etc.) • Utilizarea argumentată a conceptelor din informatică și tehnologia calculatoarelor în rezolvarea de probleme bine definite din ingineria sistemelor de operare și în aplicații ce impun utilizarea de hardware și software în sisteme industriale și în sisteme informatice • Rezolvarea de probleme uzuale din domeniul ingineriei sistemelor de operare folosind concepte ale științei calculatoarelor și tehnologiei informației referitoare la utilizarea de software dedicat și de mijloace de proiectare asistată de calculator (CAD) și la adaptarea și extinderea acestora • Selectarea și evaluarea în calitate de utilizator, de software dedicat și mijloace de proiectare asistată de calculator (CAD) pentru aplicații din ingineria sistemelor de operare, calculatoarelor, tehnologia informației și comunicațiilor • Folosirea proiectării hardware –software integrate și a ingineriei programării ca metodologii dedezvoltare, inclusiv în vederea unei modelări la nivel de sistem de operare. <p>CPL 3. Utilizarea fundamentelor automatizării, a metodelor de modelare, simulare, identificare și analiză a proceselor, a tehnicilor de proiectare asistată de calculator</p> <ul style="list-style-type: none"> • Explicarea și interpretarea problemelor de automatizare a unor tipuri de procese prin aplicarea fundamentelor automatizării, a metodelor de modelare, identificare, simulare și analiza proceselor, precum și a tehnicilor de proiectare asistată de calculator • Rezolvarea unor tipuri de probleme de conducere prin: folosirea de metode și principii de modelare, elaborarea de scenarii de simulare, aplicarea de metode de identificare și de analiză a unor procese (inclusiv procese tehnologice) și sisteme • Configurarea și implementare asistemelor de conducere a proceselor industriale, roboților și liniilor de fabricație flexibile, precum și alegerea echipamentelor, acordarea și punerea în funcțiune a structurilor aferente. <p>CPL 4. Proiectarea, implementarea, testarea, utilizarea și mentenanța sistemelor cu echipamente de uz general și dedicat, inclusiv rețele de calculatoare, pentru aplicații de automatizată și informatică aplicată</p> <ul style="list-style-type: none"> • Definirea cu ajutorul principiilor de funcționare și proiectare, a cerințelor standardelor aplicabile și a metodelor de implementare, testare, mentenanță și exploatare a echipamentelor folosite în aplicațiile de automatizată și informatică aplicată • Explicarea și interpretarea metodelor de proiectare, implementare, testare, utilizare și mentenanță a echipamentelor de uz general și dedicat, folosite pentru aplicații de conducere automată și de informatică aplicată.
Competențe transversale	<p>CT1. Executarea responsabilă a sarcinilor profesionale în condițiile unei autonomii restrânse și asistență calificată.</p> <p>CT2. Familiarizarea cu rolurile și activitățile specifice muncii în echipă și cu distribuirea de sarcini între membri pe nivele subordonate.</p> <p>CT3. Conștientizarea nevoii de formare continuă, utilizarea eficientă a resurselor și tehnicilor de învățare pentru dezvoltarea personală și profesională.</p>

6. Obiectivele unității de curs/modulului

Obiectiv general	Introducerea studenților în domeniul sistemelor de operare.
Obiective specifice	<ol style="list-style-type: none"> 1. Studiul structurii și funcționării sistemului de operare, introducerea principalilor algoritmi și instrumente program și tehnice de gestiune a resurselor. 2. Expunerea bazelor contemporane ale metodelor și mijloacelor de elaborare a resurselor program de sistem, inclusiv, operații asincrone, tratarea întreruperilor, interfețele S.O., compromisele dintre dispozitivele tehnice și resursele program. 3. Formarea de cunoștințe și abilități pentru analiza și compararea punctelor de vedere diferite asupra sistemelor de operare pentru luarea deciziilor competente la necesitate. 4. Pregătirea pentru proiectarea și analiza sistemelor de operare, crearea și dezvoltarea aptitudinilor de programator de sistem. 5. Dezvoltarea abilităților de lucru în echipă și colaborarea în dezvoltarea proiectelor.

7. Conținutul unității de curs/modulului

Tematica activităților didactice	Numărul de ore
	învățământ cu frecvență la zi
Tematica prelegerilor	
Tema 1. Introducere. Scopul și obiectivele cursului. Noțiuni și termeni din domeniul resurselor tehnice. Noțiuni și termeni din domeniul sistemelor de operare. Tipuri de sisteme de operare, obiective și funcții. Exemple de sisteme de operare. S.O. și procesele. Mașina extinsă și mașina ierarhică. Alte puncte de vedere asupra SO. Abordare funcțională. Abordare din punctul de vedere al interfeței cu utilizatorul. Evoluția SO. De la “poartă deschisă” la tratarea pe loturi. Multiprogramarea și partajarea timpului. Sisteme cu timp partajat. Windows, Unix și alte sisteme. Standardele sistemelor de calcul deschise. Sisteme de operare cu micronucleu.	4
Tema 2. Tehnici de execuție și comunicație. Mecanisme de execuție secvențială. Cazul procedurilor. Cazul corutinelor. Starea și contextul procesorului. Activități asincrone. Mecanisme de comutare a contextului unui procesor. Întreruperi. Devieri și apelarea supervisorului. Exemple de sisteme de întreruperi. Utilizarea devierilor și apelării supervisorului. Exemple de utilizare a întreruperilor. Programarea operațiilor de I/E. Organizarea generală. Periferice, controlere, canale. Adresarea dispozitivelor periferice. Metode de gestionare a dispozitivelor periferice. Operațiile de intrare/ieșire sincrone. Operațiile de intrare/ieșire asincrone cu întreruperi. Intrări/ieșiri buferizate în memorie.	2
Tema 3. Gestionarea activităților paralele. Exemple introductive. Noțiune de proces secvențial. Proces unic. Context. Relații între procese. Mulțimi de procese. Paralelism. Concurența proceselor. Resurse virtuale. Excludere mutuală. Sincronizarea proceselor. Monitorul – mecanism de sincronizare. Implementarea sincronizării. Probleme-tip. Administrarea unei resurse partajate. Alocarea resurselor banalizate. Modelul cititorului și redactorului. Modelul producătorului și consumatorului. Relația client-server. Administrarea intrărilor-ieșirilor. Sincronizarea temporală. Gestionarea dinamică a proceselor. Sincronizarea în Windows. Procese și fire. Administrarea obiectelor de sincronizare în Windows. Sincronizarea în MFC. Utilizarea secțiunilor critice în Windows.	2
Tema 4. Administrarea proceselor. Excluderea mutuală prin așteptare activă. Algoritmul lui Dekker. Așteptarea activă în sisteme multiprocesorale: Test & Set. Semaforul – instrument elementar pentru excluderea mutuală. Funcționarea și structura unui nucleu de sincronizare. Stările unui proces. Fire de așteptare. Administrarea contextelor și schemele primitivelor. Realizarea unui nucleu de sincronizare. Organizarea generală. Interfețele. Structuri și algoritmi. Realizarea monitoarelor. Algoritmi de bază. Tratarea întreruperilor. Tratarea erorilor. Operații asupra proceselor. Crearea și distrugerea proceselor. Suspendarea și reluarea. Excluderea	4

mutuală și alocarea procesorului. Realizarea pentru cazul monoprosesor. Realizarea pentru cazul unui sistem multiprosesoral. Procese și fire în Linux. Crearea proceselor. Distrugerea proceselor. Daemoni în Linux. Obținerea informațiilor despre procese în Linux.	
Tema 5. Gestiunea informației. Principiile gestiunii informației. Definiții generale. Interpretarea numelor. Construirea căii de acces. Structura reprezentărilor. Descriptori. Contexte și medii. Legarea. Protecția. Domenii și drepturi de acces. Problemele protecției. Desemnarea și legarea fișierelor și intrărilor-ieșirilor. Căi de acces la un fișier. Desemnarea externă a fișierelor. Cataloage. Introducere. Organizarea arborescentă. Legarea fișierelor cu fluxurile de intrare-ieșire. Legarea programelor și datelor. Etapele de viață a unui program. Funcționarea unui încărcător. Funcționarea unui editor de legături. Legarea prin substituție. Legarea prin înlănțuire. Mecanisme de gestiune a obiectelor. Segmentarea.	4
Tema 6. Gestiunea fișierelor. Noțiuni generale. Funcțiile unui sistem de gestiune a fișierelor. Organizarea generală a unui sistem de gestiune a fișierelor. Organizarea logică a fișierelor. Acces secvențial. Acces direct. Cheie unică. Chei multiple. Organizarea fizică a fișierelor. Alocare secvențială. Alocare non contiguă. Blocuri înlănțuite. Tabele de alocări. Alocarea memoriei secundare. Realizarea funcțiilor de acces elementar. Organizarea descriptorilor. Localizarea fizică. Informații de utilizare. Crearea și distrugerea fișierelor. Deschiderea și închiderea fișierelor. Acces elementar la informații în cadrul unui fișier. Securitatea și protecția fișierelor. Redundanța internă și restabilirea informațiilor. Salvare periodică a fișierelor. Autentificarea în Windows NT. SGF al sistemului de operare Unix.	2
Tema 7. Alocarea resurselor. Noțiuni generale. Definiții. Probleme în alocarea resurselor. Exemple de sisteme cu fire de așteptare. Modele de alocare a unei resurse unice. Alocarea procesorului. Introducere. Primul sosit, primul servit. Cererea cea mai scurtă este servită prima. Carouselul și modele derivate. Disc de paginare. Tratarea blocărilor. Enunțul problemei. Algoritmi de prevenire. Algoritmi de profilaxie. Algoritmii bancherului. Algoritmi de detectare și tratare.	2
Tema 8. Administrarea memoriei. Memorie virtuală. Alocarea dinamică a memoriei. Comportamentul programului într-o memorie virtuală liniară. Proprietățile lanțurilor de referințe. Comportamentul într-o memorie cu capacitate limitată. Comportamentul programelor segmentate. Metoda du-te – vino simplă. Partiționarea fixă. Partiționarea variabilă. Mecanisme de reamplasare dinamică. Algoritmi de alocare. Amplasarea și reamplasarea segmentelor. Utilizarea algoritmilor de alocare. Paginarea unei memorii liniare. Paginare simplă. Paginare la două niveluri. Paginarea unei memorii segmentate. Implementarea paginării. Mecanismul de deviere “pagină lipsă”. Partajarea și protecția informației. Parametrii unei discipline de alocare. Algoritmi de reamplasare cu partiție fixă. Algoritmi cu partiție variabilă. Instabilitatea și thrashing-ul. Algoritmii de reamplasare cu partiție variabilă. Reglarea șarjei și echilibrarea unei configurații. Memorii ierarhice. Noțiune de cache. Migrarea fișierelor. Mijloacele INTEL de susținere a segmentării memoriei. Mecanismul segmentare cu paginație.	4
Tema 9. Elaborarea unui sistem de operare. Decompoziție ierarhică și mașini abstracte. Conceptul de descendență și structuri cu nivele. Interfețe și specificări. Organizarea unui sistem mono-utilizator. Specificările și organizarea generală a sistemului. Funcțiile sistemului. Limbajul de comandă. Decompoziția sistemului. Interfețe interne. Primitive de intrare/ieșire. Intrări/ieșiri pentru unitatea de disc. Introducere – extragere caractere. Sistemul de gestionare a fișierelor. Organizarea logică. Organizarea fizică. Operații cu fișierele. Intrări-ieșiri logice și gestiunea fluxurilor. Interpretorul limbajului de comandă. Schema generală. Mediul de execuție. Încărcarea unui program. Tratarea erorilor de execuție. Partajarea unei mașini: mașini virtuale. Exemplu de elaborare a unui sistem de operare multitasking.	2
Tema 10. Sisteme de operare de rețea. Structura sistemelor de operare de rețea. Tipurile sistemelor de operare de rețea. Sisteme de operare pentru grupuri de lucru și Sisteme de operare pentru rețele în cadrul unei întreprinderi. Sisteme distribuite. Caracteristici ale sistemelor distribuite. Exemple de sisteme distribuite. Partajarea resurselor și Web-ul. World	4

Wide Web.	
Total prelegeri:	30
Tematica lucrărilor de laborator	
Laborator 1. Instalarea SO Linux	2
Laborator 2. Bazele utilizării interfeței SO GNU/Linux	2
Laborator 3. Sistemul de fișiere Linux	2
Laborator 4. Manipularea fișierelor Linux	4
Laborator 5. Procesarea fluxurilor textuale în SO GNU/Linux	2
Laborator 6. Programarea intrărilor-ieșirilor	2
Laborator 7. Programarea componentelor de administrare a procesorului	2
Laborator 8. Programarea componentelor de administrare a informației	2
Laborator 9. Programarea componentelor de administrare a memoriei	2
Laborator 10. Elaborarea unui sistem de operare propriu	12
Total lucrări de laborator:	30

8. Referințe bibliografice

Principale	<ol style="list-style-type: none"> 1. Charles Petzold. Programing Windows. Fifth Edition. The definitive guide to the Win32 API. Microsoft Press, 2002, 1105 pp. 2. Charles Petzold. Programarea în Windows 95, - Editura Teora, București, 1996, 1062 pp. 3. Ciclu de prelegeri la disciplina “Programarea în Windows”, Varianta de calculator. UTM, bloc 3, 506 4. V.Beșliu ș.a. Îndrumar metodic pentru lucrări de laborator la disciplina “Programarea în Windows”, UTM, 2006 5. Doru Turturea. Programarea aplicațiilor Windows în limbajul C. Editura Tehnică, București, 1995, 540 pp. 6. Peter Norton, Paul Yao. Windows 3.1 Advanced Programming Techniques. – Sybex, 1992 7. Florica Moldoveanu, Gabriel Hera. Programarea aplicațiilor Windows, - Editura Teora, București, 1994
Suplimentare	<ol style="list-style-type: none"> 1. Stephen Ferg. Event-Driven Programming: Introduction, Tutorial, History. http://Tutorial_EventDrivenProgramming.sourceforge.net (08.07.2017) 2. Bertrand Meyer. TOUCH OF CLASS. Learning to Program Well with Objects and Contracts. Sptinger, http://touch.ethz.ch/ (08.07.2017)

9. Evaluare

Curentă		Proiect de an	Examen final
Evaluarea 1	Evaluarea 2		
30%	30%	-	40%
Standard minim de performanță			
Prezența și activitate la prelegeri și lucrări de laborator;			
Obținerea notei minime de „5” la fiecare dintre evaluări și lucrări de laborator;			