

REZUMATELE FIȘELOR DISCIPLINELOR_AN IV_TI

I. Disciplina: **Sisteme Distribuite**

II. Statutul disciplinei: ☒ obligatoriu ☐ opțional ☐ facultativ

III. Precondiții

(Fiabilitate, inginerie software (nivel mediu) și programarea calculatoarelor Java – nivel mediu, programare orientată pe obiecte-POO. Arhitectura sistemelor de calcul.)

IV. Conținutul disciplinei: Introducere – tehnologii client-server și concepte de bază privind proiectarea și implementarea de servicii (client-server) (1) Tehnologii client-server; (2) Caracteristicile serviciilor distribuite; (3) Comunicația cu servicii distribuite. Arhitectura sistemelor distribuite (1) Modele de sisteme distribuite. (2) Probleme de proiectare (3) Implementare aplicații client-server în sisteme distribuite utilizând socket. Fire de execuție în sisteme distribuite – partea I (1) Elemente generale. (2) Clasa Thread și interfața Runnable cu execuție Java în sisteme distribuite. (3) Stările firelor de execuție. (4) Sincronizarea firelor de execuție – acces concurent la resurse și cooperare producător-consumator de resurse. Fire de execuție în sisteme distribuite – partea II (1) Deadlock. (2) Grupuri de Thread-uri. (3) Execuție multithread în sisteme distribuite – implementare în Java. Serializare și clonare în sisteme distribuite (1) Definiții, probleme, avantaje (2) Metode de realizare a serializării folosind Java și verificarea compatibilității (3) Serializare grafuri de obiecte (4) Clonarea obiectelor în sistemele distribuite folosind Java. Invocarea de metode la distanță (RMI) în sisteme distribuite – partea I (1) Definiții, arhitectură, etape în utilizarea RMI (2) Clase, interfețe și utilitare pentru utilizarea RMI (3) Transferul de argumente și încărcarea dinamică a claselor de obiecte. Invocarea de metode la distanță (RMI) în sisteme distribuite – partea II (1) Transferul de comportament server-client și client-server (2) Activarea obiectelor la distanță (3) Colectarea memoriei (4) Comparatie RMI - CORBA. Servicii WEB (1) Arhitecturi orientate pe servicii (SOA) (2) Concepte generale pentru servicii Web (3) Standarde și protocoale pentru servicii Web (4) Protocolul de acces SOAP. (5) Limbajul WSDL. Arhitecturi paralele. Algoritmi de procesare paralelă (1) Istoric, clasificare, modele de arhitecturi paralele. (2) Analiza performanțelor arhitecturilor paralele. (3) Scalabilitatea sistemelor distribuite. (4) Modelul abstract PRAM (5) Cerințele algoritmilor paraleli (6) Exemple de execuție a algoritmilor paraleli. Standardul MPI de comunicație prin mesaje în sistemele paralele și distribuite. (1) Generalități. (2) Comunicatori și funcții MPI. (3) Topologii virtuale. (4) Comunicarea între procese. (5) Sincronizarea proceselor. (6) Bariere și operații colective. (7) Exemple de lucru. Comunicația punct-la-punct (peer-to-peer) de tip client-server și instanță-instanță (1) Protocoale UDP și TCP. (2) Comunicația peer-to-peer. (3) Protocoale alternative la UDP/TCP. (4) Soluții fezabile pentru dezvoltarea de servicii. (5) Exemplu de lucru cu socket API. (6) Java NIO. (7) Exemplu cu SCTP Sockets în Java. Comunicația cu servicii externe și alte instanțe ale aceluiași serviciu (1) Comunicația cu servicii externe. (2) Comunicația cu alte instanțe ale aceluiași serviciu. (3) Comunicația de tip multicast. (4) Topologii arborescente. Primirea și procesarea cererilor în sisteme distribuite (1) Etape de procesare. (2) Elemente comune și diferite ale serviciilor distribuite. (3) Planificarea thread-urilor pe core-uri de procesoare. Design-ul arhitecturii software a unui serviciu în limbajul Java (1) Serviciu pe o singură mașină cu procesare asincronă a cererilor. (2) Mixturi de fire de execuție folosind Java. Configurarea, implementarea automata și testarea serviciilor distribuite (1) Configurare serviciilor distribuite. (2) Implementarea automata a serviciilor distribuite. (3) Testarea serviciilor distribuite.

V. Bibliografia minimală obligatorie

- 1) Maarten van Steen, Andrew S. Tanenbaum, "Distributed Systems", ediția 3, versiunea 01, ISBN: 978-90-815406-2-9 (versiunea digitală), 2017
- 2) Oracle Distributed Systems, Charles Dye, Publisher: O'Reilly, First Edition April 2014, ISBN: 1-56592-432-0, 548 pages
- 3) Mihai Horia Zaharia, "Sisteme Paralele și distribuite", Ed "Gh. Asachi", Iași, 2013, ISBN 973-621-056
- 4) <http://www.cdk5.net>, George Coulouris, Jean Dollimore, Tim Kindberg and Gordon Blair, *Distributed Systems Concepts and Design (5th ed.)*, Addison Wesley, 2011

VI. Fond de timp alocat pe forme de activitate

Semestrul	Forme de activitate/ număr de ore				Număr de credite
	Curs	Seminar	Laborator	Proiect	
	3x14=42	-	1x14=14	-	4

VII. Procedura de evaluare a cunoștințelor: Examen

I. Disciplina: INSTRUMENTE PENTRU DEZVOLTAREA PROGRAMELOR

II. Statutul disciplinei: ☒ obligatoriu ☐ opțional ☐ facultativ

III. Precondiții: Programare Orientata Pe Obiecte, Programare Structurata sau Procedurala

IV. Conținutul disciplinei:

1. Introducere - care este obiectul de studiu al disciplinei Medii de programare și ce loc ocupă
2. Date simple
3. Date structurate I: Vectori și matrice, liste sau colecții
4. Date structurate I: Structuri, obiecte, clasa, imagine
5. Elementele programării structurate
6. Subprograme
7. Noțiunile de bază ale programării vizuale
8. Exemple de programare în mediu vizual
9. Conceptele programării orientate pe obiecte 4 ore

V. Bibliografia minimală obligatorie

1. Donald E. Knuth- Arta programării calculatoarelor, vol. 1-3, Teora, București.
2. ***Microsoft Visual Basic 6.0 Ghidul programatorului, Ed. Teora.
3. John W Rittinghouse, A Beginner's Guide to Gambas, e-book, Copyright 2005, John W Rittinghouse – disponibilă on-line.
4. B. Pătruț – Aplicații în VB, Teora, 2000.
5. Daniel Campos, Jose Luis Redrejo, Gambas – Programacion visual con Software Libre, e-book – disponibilă on-line, 2009.
10. GAMBAS site: [Http://gambas.sourceforge.net](http://gambas.sourceforge.net)

VI. Fond de timp alocat pe forme de activitate

Semestrul	Forme de activitate/ număr de ore				Număr de credite
	Curs	Seminar	Laborator	Proiect	
I	2 × 14 = 28		1 × 14 = 14		4

VII. Procedura de evaluare a cunoștințelor: Examen

III. Disciplina: **Sisteme Tolerante la defecte**

IV. Statutul disciplinei:

☒

obligatoriu

☐

opțional

☐

facultativ

III. Precondiții

(Fiabilitate, inginerie software (nivel mediu) și programarea calculatoarelor Java – nivel mediu, programare orientată pe obiecte-POO. Arhitectura sistemelor de calcul.)

IV. Conținutul disciplinei: Principii generale privind sistemele tolerante la defecte. (1) Structura sistemului și implementarea toleranței la defecte. (2) Modelarea conceptului de toleranță la defecte. (3) Strategii de implementare a toleranței la defecte. (4) Perspective privind dezvoltarea sistemelor tolerante la defecte. Concepte și definiții fundamentale caracteristice domeniului testării. Diagnostibilitate. Strategii de implementare a toleranței la defecte. Fiabilitatea sistemelor de calcul (1) Scopuri ale toleranței la defectare. (2) Definiții (3) Aplicații ale sistemelor tolerante la defectare; (a) Sisteme cu funcționare îndelungată (b) Aplicații critice de calcul (c) Aplicații necesitând: Amânarea mentenanei și Disponibilitate ridicată. (4) Toleranța la defectare ca obiectiv de proiectare (a) Definiții (b) Cauzele și caracteristicile defectelor. Utilizarea redundanței ca abordare a toleranței la defecte (1) Redundanța fizică; (a) Replicare activă (b) Backup primar. (2) Redundanța informațională; (a) Codificarea datelor (b) Coduri de paritate (c) Sume de verificare (d) Coduri M-din-N (e) Coduri ciclice de codificare (f) Coduri aritmetice. Analiza unor tehnici de implementare a toleranței la defecte. Tehnici de detecție a erorilor (1) Considerații critice asupra tehnicilor de implementare a toleranței la defecte. (2) Tehnici de reconfigurarea a sistemelor la apariția defectărilor. (3) Model al reconfigurării în sistemele multiprocesor. (4) Arhitectura unor sisteme tolerante la defecte. (5) Abordarea concurentă a defectelor. Modelarea defectelor (1) Definiții: fault, error, failure (2) Evitarea defectelor VS tolerarea defectelor (3) Modele de blocare logică (4) Detectabilitate și redundanță (5) Echivalența și localizarea defectelor (6) Dominanță. (7) Modelarea erorilor. (8) Modelarea cu modele Markov și lanțuri Markov discrete. Generarea vectorilor stimul-test pentru scheme combinate: (1) Metoda activării unei cai (2) Metoda Roth (3) Metoda Poage (4) Elaborarea experimentelor de testare pentru scheme secvențiale (5) Design for testability. Analiza indicatorilor de fiabilitate, disponibilitate și mentenabilitate (1) Definiții (2) Fiabilitatea sistemelor serie, paralel și mixte (3) Sisteme duplex tolerante la defecte (4) Disponibilitatea sistemelor de calcul (5) Mentenabilitatea sistemelor de calcul (6) Aplicații. Toleranța la defectare (1) Tehnici de obținere a redundanței software și hardware (2) Designul sistemelor tolerante la defecte (3) Injectia de erori (4) Sisteme tolerante la defecte. (5) Rețele și sisteme distribuite tolerante la defecte; (a) Heartbeats (b) Timere Watchdog. Metode de proiectare pentru combaterea defectelor. Tehnici de proiectare pentru realizarea toleranței la defectare (1) Conceptul de redundanță (2) Redundanță hardware. Structuri redundante pentru implementarea toleranței la defecte în sistemele hardware (1) Viteza de apariție a defectelor hardware (2) Considerații asupra utilizării redundanței protective în sisteme. (3) Scheme de implementare a redundanței la nivel hardware: (a) Structuri redundante protective statice de tip individual și global rezultate prin multiplicare. (b) Structura redundantă logică majoritară (c) Structura redundantă protectivă statică cu logică cuadruplă / (d) cu logică prin cablare. (e) Structura redundantă prin codare (f) Structura redundantă protectivă dinamică (g) Structura redundanță hibridă. (4) Structuri redundante pentru interconexiunile unui sistem. (5) Probleme de sincronizare în sistemele digitale cu structură redundantă. (6) Criterii de comparare pentru structurile redundante. (a) Indici de îmbunătățire a fiabilității sistemelor cu structură redundantă. (b) Indici de cost și eficiență. (c) Studii de caz privind evaluarea performanțelor unor sisteme cu structură redundantă. Sisteme tolerante la defecte cu redundanță modulară (1) Sisteme M-din-N redundante; (a) Evaluarea fiabilității (b) Circuite de votare. (2) Variații ale redundanței N-modulare; (a) Redundanța modulară la nivelul unităților (b) Redundanța dinamică. (3) Redundanța hibridă. (4) Redundanța modulară cu filtrare. Testarea sistemelor software. Sisteme software embedded. Sisteme digitale total autotestabile. Reconfigurarea sistemelor la apariția defectelor. Model de reconfigurare în sisteme multiprocesor (1) Teste de acceptare. (2) Încapsulările. (3) Reîntinerirea software-ului (4) Toleranță la defecte software bazată pe algoritmi. Defecte specifice tehnicii de calcul (a) Defecte de blocare (stuck-at faults) (b) Modelul de scurt-circuit (c) Gate Oxide Shorts (d) Defecte temporare (e) Defecte tranzitorii (f) Defecte permanente

Perturbatii în sistemele de calcul (a) Consideratii generale (b) Perturbatii datorate diafoniei (c) Perturbatii datorate reflexiilor. Managementul testării: planificarea activităților de testare, dezvoltarea mediului de testare. Evaluarea testării. Redundanța temporală (Checkpoint). Studii de caz. (1) Sistemele NonStop de la TandemComputers. Arhitectura, software, mentenanța (2). Sisteme Stratus (3) Sisteme IBM: G5 și Sysplex (4) Intel Itanium (5) Subsistemele Cassini.

V. Bibliografia minimală obligatorie

1. Băjenescu, T.I. Fiabilitatea, disponibilitatea și mentenabilitatea sistemelor electronice complexe, Editura de Vest 2017
- 2) Mihalache, A. Când calculatoarele gresesc. Fiabilitatea sistemelor de programe (software), Ed.Didactică și pedagogică, București 2015
- 3) Popovici, Al.A. Proiectarea securității sistemelor complexe, Ed.Stiintifică și enciclopedică, București 1988
- 4) Ștefănescu, C. Sisteme tolerante la defecte, Matrix Rom, București 2017
- 5) Israel Koren, C. Mani Krishna, *Fault Tolerant Systems*, ediția 2, Morgan Kaufmann, eISBN: 9780128181065, 21.09.2020.

VI. Fond de timp alocat pe forme de activitate

Semestrul	Forme de activitate/ număr de ore				Număr de credite
	Curs	Seminar	Laborator	Proiect	
	2x14=28	-	1x14=14	-	4

VII. Procedura de evaluare a cunoștințelor: colocviu

I. Disciplina: Informatică Industrială

II. Statutul disciplinei: ☐ obligatoriu ☒ opțional ☐ facultativ

III. Precondiții (precizarea eventualelor cerințe de cunoaștere/promovarea prealabilă a unor alte discipline)

IV. Conținutul disciplinei: 1 Introducere în informatica industrială. Elemente generale despre controlerelor. Structura controlerului programabil. Funcționarea controlerului programabil. Controlerul compact PS 4-201-MMM1. Descrierea echipamentului. Crearea unui proiect în SUCO SOFT. Crearea fișierului de configurare. Scrierea programului. Setarea parametrilor specifici programului. Transferul programului pe controlerul PS 4-201-MM1. Forțarea ieșirilor. Elementele unei instrucțiuni. Tipuri de date. Intrări. Ieșiri. Instrucțiuni de încărcare și transfer. Instrucțiuni logice. Instrucțiuni de salt. Operatori de comparație. Apelul și revenirea din blocurile funcționale. Instrucțiuni aritmetice. Blocuri funcționale TON, TOF. Bloc funcțional TimeGenerator. Generator de tren de impulsuri. Numărătoare pe 16 biti. Programarea pe baza diagramelor Ladder. Blocul funcțional de control secvențial SFC. Module externe EM 4. Rețele cu SUCOnet K.

V. Bibliografia minimală obligatorie

- Culea George, Informatică industrială – Note de curs – laborator, Universitatea „Vasile Alecsandri” din Bacău 2019;
- SUCOSOFT S40 Programming Software, EATON, 2014;

VI. Fond de timp alocat pe forme de activitate

Semestrul	Forme de activitate/ număr de ore				Număr de credite
	Curs	Seminar	Laborator	Proiect	
7	2x14=28		2x14=28		4

VII. Procedura de evaluare a cunoștințelor Examen.

I. Disciplina: Programare WEB

II. Statutul disciplinei: ☐ obligatoriu ☒ opțional ☐ facultativ

III. Precondiții:

IV. Conținutul disciplinei:

1. Internetul și Caracteristicile Web-ului. Clienti & servere Web. Arhitectura unei aplicații web. Modele de formatare a documentelor Web.
2. Structura unui document HTML. Taguri de baza, tabele, formulare. Crearea paginilor web utilizând limbajul (X)HTML, Taguri (X)HTML, Structura documentelor HTML5. Formulare web. Elemente multimedia pentru programarea web. DHTML
3. Stiluri. Conținut și design. Formatarea paginilor Web prin intermediul stilurilor CSS (Cascading Style Sheets), (CSS2 & CSS3).
4. Servicii, sisteme și medii de dezvoltare a aplicațiilor în internet. XML
5. Realizează transferul de informații, adresarea resurselor în spațiu www și protocoale web (protocolul HTTP și Metodele GET, POST). Servere Web (HTTP), Servere de Aplicații Web Java: (Java Servlets. JSP (Java Server Pages), ASP (Active Server Pages)) (concepțe, caracteristici, exemple). Browsers Web;
6. Programare web la nivel de client-side - Concepțe și tehnici de baza. Tehnologii de dezvoltare și implementare. JavaScript (js). DOM- obiect Object Model. Operatori și expresii. Obiecte (Obiectul "Window" și de nivelul trei).
7. Ajax. jQuery. Expresii Regulate (ER)(Regular expressions). Scripturi CGI CGI (Common Gateway Interface). Scripturi scrise în limbaje compilate.
8. Programare web la nivel de server (server-side)- Tehnologii (avansate) de dezvoltare și implementare. Limbaje de scripting-PHP: Elemente de baza.
9. Tehnici de accesare a bazelor de date (MySQL) folosind PHP și a limbajului SQL; Conectarea la diferite BD. ODBC.
10. Tranzacții web. Cookie-uri & Sesiuni. Elemente de securitate și optimizare a aplicațiilor web. Securitatea Clientului Web. SQL injection. Roboti, Administrarea conținutului site-urilor Web: Optimizarea site-urilor Web pentru indexarea de către motoarele de căutare. Strategii SEO-Search Engine Optimization.
11. Web semantic. Publicare site/bazelor de date pe WEB. Organizarea și întreținerea siteurilor WEB. Arhitectura orientată spre servicii web (web services):
12. SOA- Service Oriented Architecture (Descrierea serviciilor Web,
13. Limbajul WSDL - Web Services Description Language;
14. Protocolul SOAP - Simple Object Access Protocol);

V. Bibliografia minimală obligatorie

- [1] Cursuri și laboratoare: <http://examene.pe.ub.ro/> ; <http://cadredidactice.ub.ro/pruteanue/>
- [2] Traian Anghel – Dezvoltarea aplicațiilor web folosind PHP și AJAX, EduSoft, 2007
- [3] Alexandru Deva – Programarea web altfel, EduSoft, 2007

VI. Fond de timp alocat pe forme de activitate

Semestrul	Forme de activitate/ număr de ore				Număr de credite
	Curs	Seminar	Laborator	Proiect	
I	2 × 14 = 28		2 × 14 = 28		4

VII. Procedura de evaluare a cunoștințelor: examen

I. Disciplina: Proiectare Software

II. Statutul disciplinei: ☒ obligatoriu ☐ opțional ☐ facultativ

III. Precondiții: Promovarea disciplinelor: Electronică digitală, Electrotehnică și electronică

IV. Conținutul disciplinei:

1. Introducere în proiectarea software. Motivație și concepte generale de proiectare. Privire de ansamblu asupra proiectării software. Aspecte administrative
2. Elemente fundamentale privind proiectarea software. Aspecte generale privind UML. Stabilirea aplicației țintă, a temei de proiectare
3. Elemente fundamentale privind proiectarea software. Modelarea structurală în UML. Modelarea comportamentului în UML. Stabilirea cerințelor de proiectare
4. Fundamentele arhitecturii software. Fundamentele ingineriei cerințelor. Proiectarea arhitecturii software
5. Stiluri și șabloane arhitecturale. Elemente generale și istoricul stilurilor și șabloanelor arhitecturale. Sisteme centrate pe date și sisteme bazate pe fluxuri de date
6. Stiluri și șabloane arhitecturale. Sisteme distribuite. Sisteme interactive și ierarhice
7. Proiectarea de detaliu. Privire generală. Proiectarea și comportamentală a componentelor. Principii de proiectare
8. Folosirea șabloanelor creaționale în proiectarea de detaliu. Abstract Factory. Factory Method
9. Folosirea șabloanelor creaționale în proiectarea de detaliu. Builder. Prototype. Singleton.
10. Folosirea șabloanelor structurale în proiectarea de detaliu. Adapter. Composite. Façade
11. Folosirea șabloanelor comportamentale în proiectarea de detaliu. Iterator. Observer.
12. Proiectarea de construcție. Proiectarea de construcție bazată pe fluxuri, stări și tabele. Limbaje pseudocod, stiluri și evoluția calității.
13. Gestiunea procesului de proiectare software. Un framework pentru gestiunea proiectării software. Planificarea
14. Gestiunea procesului de proiectare software. Implementarea și terminarea. Recapitulare finală. Recapitularea finală și evaluarea proiectelor

V. Bibliografia minimală obligatorie

- OTERO, C.E.: Software Engineering Design, CRC Press, 2012.
site: <http://softwareengineeringdesign.com/Default.htm>
- BASS, L., CLEMENTS, P., KAZMAN R.: Software Architecture in Practice, 2nd ed., Addison-Wesley, 2013
- KRUCHTEN, PH.: Architectural Blueprints – The 4+1 View Model of Software Architecture, IEEE Software 12 (6), 2015, pp. 42-50.
- SHAW, M.: The Coming-of-Age of Software Architecture Research, in Proc. of the 23rd ICSE, IEEE Comp. Soc. 2001, 656
- SHAW, M., GARLAN, D.: Software Architecture: Perspectives on an Emerging Discipline, Prentice-Hall, 2016.

VI. Fond de timp alocat pe forme de activitate

Semestrul	Forme de activitate/ număr de ore				Număr de credite
	Curs	Seminar	Laborator	Proiect	
I	2 × 14 = 28		2 × 14 = 28		4

VII. Procedura de evaluare a cunoștințelor: colocviu

I. Disciplina: *Proiectarea cu microprocesoare*

II. Statutul disciplinei: ☒ obligatoriu ☐ opțional ☐ facultativ

III. Precondiții *matematică, electronică digitală, proiectare logică*

IV. Conținutul disciplinei: *Programarea în limbaj de asamblare. Programarea microprocesorului intel 8086. Programarea microprocesorului TMS 320F240. Programarea microcontrolerelor de tip pic12, pic16 și pic 18*

V. Bibliografia minimală obligatorie

1. Rotar Dan, Microprocesoare, Curs digital, 2014
2. Vințan N. Lucian, Predicție și speculație în microprocesoarele avansate, Matrix Rom, București, 2002
3. Baci R., Programarea în limbaje de asamblare, Editura ALMA MATER, Sibiu, 2003.
4. Florea A., Vințan L., Simularea și optimizarea arhitecturilor de calcul în aplicații practice, Editura MATRIX ROM, București, 2003.

VI. Fond de timp alocat pe forme de activitate

Semestrul	Forme de activitate/ număr de ore				Număr de credite
	Curs	Seminar	Laborator	Proiect	
5, 6	2x14=28	-	1x14=14	2 x 14=28	7

VII. Procedura de evaluare a cunoștințelor: examen

I. Disciplina: *Fiabilitate software*

II. Statutul disciplinei: ☒ obligatoriu ☐ opțional ☐ facultativ

III. Precondiții

IV. Conținutul disciplinei:

1. Introducere
2. Elemente de teoria probabilităților-
3. Estimarea parametrilor funcțiilor de repartiție
4. Modele structurale utilizate în fiabilitate
5. Metode de calcul utilizate în fiabilitate 4ore
6. Redundanta, toleranta la defectari

7. Mentenanța sistemelor informatice

V. Bibliografia minimală obligatorie

1. Grigore Roxana, Fiabilitate, suport de curs electronic., 2016
2. Ioan P.VIZITEU, Fiabilitatea instalațiilor de protecție și automatizare utilizate în sistemele electroenergetice, 2009
3. Costica Nitu, Sisteme de conducere cu calculatoare. Fiabilitatea factorului uman, 2010
4. Martin L. Shooman, Reliability of Computer Systems and Networks, 2003

VI. Fond de timp alocat pe forme de activitate

Semestrul	Forme de activitate/ număr de ore				Număr de credite
	Curs	Seminar	Laborator	Proiect	
I	1x14=14	1x14=14			3

VII. Procedura de evaluare a cunoștințelor: *colocviu*

I. Disciplina: *Sisteme Inteligente*

II. Statutul disciplinei: ☒ obligatoriu ☐ opțional ☐ facultativ

III. Precondiții

IV. Conținutul disciplinei:

Strategii de căutare. Căutare neinformată și informată. Satisfacerea constrângerilor. Exemple: tic-tac-toe, dezvoltarea arborelui de joc, evaluarea stărilor, metodele MIN-MAX și ALPHA-BETA. Reprezentarea cunoașterii. Ierarhia cunoașterii, Rețele semantice (rețele descriptive, rețele evenimentțiale), cadre, scenarii, ontologii. Metode de optimizare (1). Algoritmi evolutivi. Optimizare uni- și multiobiectiv. Metode de optimizare (2). Evoluție diferențială, inteligența roiurilor. "Alpinism" (hill climbing). Calire simulată. Metode de inferență în logica propozițională și cea predicativă. Metode de planificare. Rețele bayesiene. Tehnici de clasificare. Învățare supervizată. Învățarea supervizată și nesupervizată. Învățare cu întărire și învățare profundă. Sisteme de calcul inteligent. Paradigma conexionistă și calcul neuronal. Elemente de bază ale unei Rețele Neuronale Artificiale. Rezolvarea problemelor de asociere - aproximare și predicție. Considerații generale asupra Sistemelor Inteligente cu logică nuanțată (SILN). Mulțimi fuzzy. 1. Mulțimi și Operații cu mulțimi fuzzy. 2. Generalizări ale operațiilor fuzzy: t-norme și s-norme. Sisteme neuro-fuzzy. Sisteme cu Logică Fuzzy (SLF). Mulțimi Fuzzy (MF) & Logica Fuzzy (LF). Principiul extensiei: (1) Definirea principiului extensiei. (2). Mulțimi fuzzy de ordinul 2 și de ordinul m. Operații cu mulțimi fuzzy de ordinul 2. Tipuri de Sisteme inteligente cu Logică Fuzzy. Sisteme de Baze de Date inteligente (FSQL-Fuzzy Structured Query Language sau Fuzzy SQL). Relații fuzzy. (1) Relații fuzzy și Operații cu relații fuzzy. (2) Compunerea relațiilor fuzzy. Compoziția max-min și max-star. Proprietăți (3) Inferența fuzzy (1) Variabile lingvistice. (1). Fuzzificarea și defuzzificarea. (2) Arhitecturi de FLC.(3) Elemente de aritmetică fuzzy. (1) Numere fuzzy și Operații cu numere fuzzy. Sisteme inteligente bazate pe cunoștințe (SIBC). Sisteme Expert (SE) cu logică fuzzy (FES - Fuzzy Expert Systems).) Sisteme evoluționiste bazate pe Algoritmi Genetici (AG). Sisteme Inteligente de Suport Decizional – SISD – (sau de asistare a deciziilor). Sisteme bazate pe Agenți (inteligenți) – proprietăți, clasificări. Sisteme Inteligente de inspirație naturală (din natură). Sisteme Inteligente Adaptive și Colaborative

V. Bibliografia minimală obligatorie

1. Russell, S. J., Norvig, P. (2010). *Artificial Intelligence: A Modern Approach*, Prentice Hall, 3rd Edition.
2. R.G. J. Klir and B. Yuan, *Fuzzy Sets and Fuzzy Logic: Theory and Applications*. Prentice Hall PTR, 2005.
3. Hariton Costin, *Sisteme Inteligente*, suport de curs, 2020.

VI. Fond de timp alocat pe forme de activitate

Semestrul	Forme de activitate/ număr de ore				Număr de credite
	Curs	Seminar	Laborator	Proiect	
I	2x14=28	1x14=14	1x14=14		4

VII. Procedura de evaluare a cunoștințelor: examen

I. Disciplina: APLICAȚII PENTRU DISPOZITIVE MOBILE

II. Statutul disciplinei: ☐ obligatoriu ☐ opțional ☒ facultativ

III. Precondiții:

IV. Conținutul disciplinei:

1. Elemente introductive despre programarea dispozitivelor mobile. Tipologii de aplicații pentru dispozitive mobile. Responsiveness.
2. Arhitectura aplicațiilor mobile Android, IOS și Windows. Exemple Xamarin Forms
3. Stocarea persistentă a datelor. Apelul serviciilor web, utilizarea protocoalelor JSON și SOAP
4. Dezvoltarea interfețelor utilizator multiplatformă. Scalabilitatea interfețelor utilizator pe ecrane de dimensiune și rezoluție diferită.
5. Capabilități. Acces la senzorii dispozitivelor și la resursele de (tele)comunicații în rețele WIFI și 4G.
6. Grafică și multimedia pe platformele Android, Windows Phone, Windows 10 și IOS.
7. Aplicații mobile multiplatformă (cross-platform)

V. Bibliografia minimală obligatorie

1. Paul POCATILU, Ion IVAN, Adrian VIȘOIU, Felician ALECU, Alin ZAMFIROIU și Bogdan IANCU - Programarea aplicațiilor Android, Editura ASE, București, 2015, ISBN: 978-606-505-856-9
2. Charles PETZOLD - Cross-platform C# programming for iOS, Android, and Windows, Microsoft Press, 2016, ISBN: 978-1-5093-0297-0

VI. Fond de timp alocat pe forme de activitate

Semestrul	Forme de activitate/ număr de ore				Număr de credite
	Curs	Seminar	Laborator	Proiect	
I	2 × 14 = 28		1x14=14		3

VII. Procedura de evaluare a cunoștințelor: colocviu

I. Disciplina: Sisteme de recunoaștere a formelor

II. Statutul disciplinei: ☒ obligatoriu ☐ opțional ☐ facultativ

III. Precondiții

IV. Conținutul disciplinei:

Recapitularea notiunilor de algebraă, statistica și probabilități. Decrierea generală a unui (SRF).

Reprezentarea imaginilor digitale funcții și cuvinte picturale, arbori quad și binari.

Abordări statistice: (1) Clasificatori de tip Bayes (și pentru clase cu distribuție normală); (2) Estimarea densității: folosind metode parametrice/ nonparametrice.

Abordări bazate pe funcții liniar discriminante. (1) Metode liniar discriminante generale. (2) Analiza discriminant liniară.

Spații de culoare: (1) Conversii (RGB – CIE XYZ - CIE-Lab, Luv, HSV, ...) (2) Compunerea imaginilor

Îmbunătățirea imaginilor. Operațiuni punctuale: *Accentuarea contrastului, Reducerea zgomotului, Binarizarea imaginilor, Negativarea imaginilor, Operațiuni de tip fereastră, Extragerea unui bit, Compresia de contrast, Scaderea imaginilor, Modelarea imaginilor prin histograme.*

Operațiuni spațiale: Mediere și filtrare. *Accentuarea contururilor, Inversarea contrastului și scalare statistică, Dilatarea imaginilor, Îmbunătățirea și Pseudocolorarea imaginilor biomedicale*

Operații integrale. (1) Transformări unitare bidimensionale discrete. (2) Restaurarea imaginilor

Transformări ale imaginilor: Determinarea conturului, Scheletizare, Subtiere.

Transformări morfologice pentru imagini **alb-negru**.

(1) **Transformări de bază:** translație, eroziune, dilatare, proprietăți (iteratie, incluziune, dualitate, ...);

(2) **Transformări morfologice compuse:** potrivirea, deschiderea, închiderea, determinarea conturului, subțierea, îngroșarea, scheletul, curățare.

Transformări morfologice ale imaginilor cu **nuante de gri**.

(1) **Transformări de bază:** eroziune, dilatare,

Transformări derivate: deschiderea, închiderea, potrivirea, determinarea conturului, subțierea, curățarea.

Transformări morfologice pentru **imagini color**: (1) Transformările elementare, (2) Transformări compuse.

Clasificare pe baza de vectori suport. Metode de recunoaștere a formelor prin ansamblu de clasificatori.

Evaluarea performanței clasificatorilor

Prelucrarea imaginilor Stereo (Anaglifo, Stereograme): accentuarea/diminuarea componentelor culorilor (anaglifo), suprapunerea imaginilor (combinarea imaginilor)

V. Bibliografia minimală obligatorie

- 1 Cursuri și laboratoare: <http://examene.pe.ub.ro/> ; <http://cadredidactice.ub.ro/pruteanue/>
- 2 C. Bishop, "Pattern Recognition and Machine Learning", Springer, 2017
- 3 Jain A.K., Fundamentals of Digital Image Processing, Prentice-Hall, London, 2017.
- 4 Pavlidis, T., Algorithms for Graphics and Image Processing. Springer-Verlag, 2014.
- 5 Vlaicu A., Prelucrarea digitală a imaginilor, Editura Albastra, Cluj-Napoca, 2017.
- 6 Giardina C R and Dougherty E R, Morphological Methods in Image and Signal Processing, Prentice-Hall, Englewood Cliffs, New Jersey, 2018.

VI. Fond de timp alocat pe forme de activitate

Semestrul	Forme de activitate/ număr de ore				Număr de credite
	Curs	Seminar	Laborator	Proiect	
2	2 x 12 = 24	1 x 12 = 12	2 x 12 = 24	-	4

VII. Procedura de evaluare a cunoștințelor: examen

I. Disciplina: Managementul proiectelor software

II. Statutul disciplinei: ☒ obligatoriu ☐ opțional ☐ facultativ

III. Precondiții:

IV. Conținutul disciplinei:

- Prezentarea generală a teoriei manageriale și a managementului proiectelor: Prezentarea generală a managementului proiectelor: Conceptul de management. Funcțiile managementului. Scurt istoric al teoriei manageriale. Definirea, caracteristicile și tipologia proiectelor. Definirea, scopul și sarcinile principale ale managementului de proiect. Importanța managementului de proiect. Părțile implicate în proiect. Ciclul de viață al unui proiect. Procesele de bază specifice proiectelor. Domeniile (zonele) de cunoaștere ale managementului de proiect
- Aspecte specifice managementului proiectelor: Conținutul și obiectivele proiectului. Structuri organizatorice utilizate în cadrul managementului de proiect. Structura descompunerii lucrărilor (WBS). Planificarea proiectelor. Tehnici de planificare. Derularea proiectelor. Finalizarea (închiderea) proiectelor. Evaluarea proiectelor

V. Bibliografia minimală obligatorie

1. Drob, C. – “Management de proiect.”, Editura Alma Mater, Bacău, 2010;

VI. Fond de timp alocat pe forme de activitate

Semestrul	Forme de activitate/ număr de ore				Număr de credite
	Curs	Seminar	Laborator	Proiect	
2	2x12==24			3x12=36	2

VII. Procedura de evaluare a cunoștințelor: colocviu

I. Disciplina: APLICAȚII INTEGRATE PENTRU ÎNTREPRINDERI

II. Statutul disciplinei: ☒ obligatoriu ☐ opțional ☐ facultativ

III. Precondiții:

IV. Conținutul disciplinei:

1. Enterprise resource planning (ERP)
2. Prezentare SAP
- 3 Limbajul de programare ABAP
- 4 Project 2010

V. Bibliografia minimală obligatorie

- 1 Amol Palekar, Bharat Patel, Shreekant Shiralkar, SAP BW 7.4—Practical Guide, SAP Press Inc, ISBN 978-1-4932-1191-3, 852 pag., 2015.
- 2 Culea George, Găbureanu Cătălin Aplicații integrate pentru întreprinderi, Editura PIM, Iași 2008, ISBN 978-606-520-059-3
- 3

VI. Fond de timp alocat pe forme de activitate

Semestrul	Forme de activitate/ număr de ore				Număr de credite
	Curs	Seminar	Laborator	Proiect	

2	2x12=24		2x12=24		3
---	---------	--	---------	--	---

VII. Procedura de evaluare a cunoștințelor: examen

V. Disciplina: Sisteme încorporate

VI. Statutul disciplinei: ☐ obligatoriu ☒ opțional ☐ facultativ

III. Precondiții (precizarea eventualelor cerințe de cunoaștere/promovarea prealabilă a unor alte discipline)

IV. Conținutul disciplinei: Introducere. Sisteme încorporate. Caracteristicile unui sistem încorporat. Sisteme încorporate hardware și software. Analiza proceselor ciclice secvențiale în vederea alegerii unei soluții de control automat. Modalități de implementare a sistemelor încorporate. Sisteme înglobate industriale. Criterii de alegere a unei soluții de control. Echipamentului TWIDO. Prezentarea echipamentului TWIDO. Variante de controlere programabile Twido. Module de extensie pentru intrări/ieșiri digitale, analogice. Module de ieșire. Protocole de comunicație. Caracteristici Controlerul modular TWDLMDA20DRT. Modul de cablare a controlerului TWDLMDA20-DRT. Modulul de intrări/ieșiri analogice TWDAMM3HT. Modul de cablare al modulului TWDAMM3HT. Funcții speciale ale controlerelor Twido. Programatoare ciclice. Editor de simboluri. Limbajul lista de instrucțiuni. Instrucțiuni pe acumulator. Instrucțiuni pe acțiuni. Instrucțiuni pe blocuri funcționale. Utilizarea parantezelor. Instrucțiuni cu stiva. Programarea cu ajutorul reprezentării Grafcet. Instrucțiuni Grafcet. Blocuri functionale de temporizare. Programarea ceasului de timp real RTC.

V. Bibliografia minimală obligatorie

- Culea George, Sisteme încorporate, Note de curs - laborator, Universitatea „Vasile Alecsandri” din Bacău 2020
- Schneider electric -TwidoSuite V2.3 Programming Guide 2011

VI. Fond de timp alocat pe forme de activitate

Semestrul	Forme de activitate/ număr de ore				Număr de credite
	Curs	Seminar	Laborator	Proiect	
8	2x14=28		2x14=28		4

VII. Procedura de evaluare a cunoștințelor Examen.

III. Disciplina: ERGONOMIE

IV. Statutul disciplinei: ☐ obligatoriu ☒ opțional ☐ facultativ

III. Precondiții (precizarea eventualelor cerințe de cunoaștere/promovarea prealabilă a unor alte discipline)

V. Conținutul disciplinei:

1. Organizarea ergonomică a muncii
2. Organizarea ergonomică a locului de muncă a unui muncitor
3. Organizarea ergonomică a locului de muncă a unui cadru de conducere
4. Analiza unor aspecte ale metodelor de muncă cu ajutorul unor metode grafice de lucru;
5. Structura procesului de muncă
6. Aparare de înregistrare a consumului de timp, a ritmului de muncă și a imaginilor folosite în organizarea ergonomică a muncii – cronociclografierea
7. Metodele folosite pentru măsurarea și studiului de timp de muncă
8. Determinarea timpului de muncă pe operații și atribuții
9. Sisteme de normative de timp de muncă pe mișcări
10. Microclimatul industrial
11. Evaluarea factorilor care determină gradul de oboseală

V. Bibliografia minimală obligatorie

- Nedeff Valentin, Panainte Mirela, Moșneguțu Emilian, 2007 – Ergonomie, Editura Alma Mater, Bacău.

VI. Fond de timp alocat pe forme de activitate

Semestrul	Forme de activitate/ număr de ore				Număr de credite
	Curs	Seminar	Laborator	Proiect	
8	3x12=36	1x12=12			3

VII. Procedura de evaluare a cunoștințelor examen.

I. Disciplina: Elaborare proiect de diplomă

II. Statutul disciplinei: ☒ obligatoriu ☐ opțional ☐ facultativ

III. Precondiții:

IV. Conținutul disciplinei:

- 1 Analiza temei lucrării de licență/Stabilirea capitolelor lucrării
- 2 Analiza gradului de aprofundare a cunoștințelor teoretice necesare realizării lucrării de licență
- 3 Studiul gradului actual al cunoașterii în domeniul temei
- 4 Sistematizarea datelor referitoare la stadiul actual al cunoașterii
- 5 Stabilirea etapelor pentru proiectarea și realizarea aplicației hardware și/sau software.
- 6 Proiectarea aplicației pe baza cerințelor temei
- 7 Testarea sau simularea aplicației
- 8 Realizarea părții practice a lucrării
- 9 Interpretarea rezultatelor
- 10 Descrierea modului de utilizare a aplicației
- 11 Verificare finală
- 12 Stabilirea concluziilor

V. Bibliografia minimală obligatorie

Bibliografia va fi stabilită împreună cu îndrumătorul de licență pentru fiecare lucrare în parte.

VI. Fond de timp alocat pe forme de activitate

Semestrul	Forme de activitate/ număr de ore				Număr de credite
	Curs	Seminar	Laborator	Proiect	
8				4x12=48	4

VII. Procedura de evaluare a cunoștințelor: colocviu

I. Disciplina: ANTREPRENORiat

II. Statutul disciplinei: ☐ obligatoriu ☒ opțional ☐ facultativ

III. Precondiții:

IV. Conținutul disciplinei:

- 1 Antreprenoriatul, Definiții, terminologie, Caracteristici, Abordări microeconomice și macroeconomice
- 2 Antreprenoriatul rural, antreprenoriatul agricol, Noțiuni de bază, termeni specifici, Caracteristicile din mediul rural, Abordarea comprehensivă
- 3 Politici în domeniul antreprenoriatului
- 4 Formarea unei culturi a antreprenoriatului
- 5 Tipologia afacerilor
- 6 Incubarea, conducerea și lichidarea afacerilor
- 7 Cumpărarea, vânzarea, franciza în afaceri
- 8 Planul de afaceri
- 9 Etica în activitatea de antreprenoriat
- 10 Responsabilitatea antreprenorului

V. Bibliografia minimală obligatorie

1. Toma S. G. (2007), Bazele economiei întreprinderii, București: Ed. ASE
2. Osborne A. E., Luecke R. (2005), Entrepreneur's Toolkit. Tools and Techniques to Launch and Grow Your New Business, Boston: Harvard Business School Press,
3. Văduva S. (2004), Antreprenoriatul, București: Ed. Economică
4. Bridge S., O' Neil K., Cromie S. (2003), Understanding Enterprise, Entrepreneurship and Small Business, New York: Palgrave, Macmillan
5. Hisrich R. D., Peters M. P. (2002), Entrepreneurship, Boston: McGraw-Hill
6. Sasu C. (2001), Inițierea și dezvoltarea afacerilor, Iași: Polirom
7. Nedeff V. 1998 – Procedee și tehnici de protecție a mediului în agricultură și industria alimentară, Ed. Tehnică Chișinău;

VI. Fond de timp alocat pe forme de activitate

Semestrul	Forme de activitate/ număr de ore				Număr de credite
	Curs	Seminar	Laborator	Proiect	
8	2x12=24			1x12=12	3

VII. Procedura de evaluare a cunoștințelor: colocviu

Structura lucrării de finalizare trebuie să cuprindă următoarele părți:

Cuprins (capitole, subcapitole, paragrafe): Introducerea în această parte se prezintă după caz: motivația alegerii temei lucrării, importanța acesteia și obiectivele lucrării.

Partea teoretică prezintă conceptele, teoriile și modelele relevante pe care se fundamentează elaborarea temei în speță. Se recomandă abordarea comparativă, critică și nu abordarea strict descriptivă. Se poate prezenta dezvoltarea istorică/evoluția a problemei. Datele trebuie să fie actuale și actualizate. Partea teoretică trebuie să fie relevantă pentru partea practică, să reprezinte suportul acesteia.

Partea practică/aplicativă (Studiu de caz, etc.) prezintă contribuțiile autorului lucrării. Contribuțiile se pot concretiza într-o cercetare cantitativă sau calitativă empirică, un proiect, un studiu de fezabilitate, prezentarea unui model, studiu experimental, studiu de caz și analiza lui, ș.a., în funcție de specificul domeniului.

Concluzii (eventual propuneri) Modul de valorificare, formularea unor opinii/observații asupra aspectelor sesizate în urma evaluării, studiului sau cercetării. Această parte vine să conchidă fuziunea cercetării teoretice cu cea aplicativă, indicând originalitatea, puterea de analiză și sinteză, într-un cuvânt fiind sinteza competențelor pe care candidatul le-a dobândit în timpul programului de studiu.

Bibliografie: Lista referințelor bibliografice utilizate la elaborarea lucrării.

Modul de redactare al lucrării de finalizare

Redactarea va respecta cerințele de realizare a lucrărilor științifice. Minimul acceptat pentru o lucrare de finalizare este de 30 pagini (fără anexe).

Lucrările vor fi editate la calculator. Formatul paginii A4, marginile de 2,5 cm. Caracterul va fi Times New Roman, mărimea de 12 pct, la 1,5 rânduri, aliniat: stanga-dreapta (justify).

Titlurile capitolelor, se numerează, mărimea 14 pct. Centrate. Bold. Titlurile subcapitolelor, se numerează, mărimea 12 pct. Aliniate la stanga.

Tabelele, figurile și graficele se includ în text, fiind numerotate și având titlu. Referirea în text la acestea se va face prin indicația: figura nr. ij sau tabelul nr. ij, unde: i – nr. capitol; j – nr. figură, tabel.

Textul va avea trimiteri fie la subsolul paginii, citate, trimiteri bibliografice în text, la sfârșitul capitolului sau lucrării.

Bibliografia poate să cuprindă diferite surse: cărți, articole, Internet. Pentru cărți se va menționa numele, prenumele autorului/autorilor, titlul, editura, anul apariției, eventual ediția, orașul, țara. Toată bibliografia prezentată trebuie să se regăsească în lucrare.

Notele de subsol trebuie să se regăsească în partea de jos a paginii indicându-se: numele, prenumele autorului/autorilor, titlul, editura, anul apariției, eventual ediția, orașul, țara și pagina unde se află informația utilizată.

Precizări finale

Lucrările de licență se vor preda atât în formă listată și îndosariată cât și sub formă electronică pe suport DVD.

Lucrările listate pe suport de hârtie nu vor conține listinguri complete ale programelor sursă sau echivalente. În lucrarea pe suport de hârtie se admit doar secvențe din fișierul sursă necesare ilustrării explicațiilor din lucrare.

Formatul electronic pe suport DVD trebuie să conțină: formatul electronic al părții scrise, fișierele sursă ale aplicației realizate, elemente auxiliare (baze de date, programe executabile, programe de test, drivere etc) necesare aplicației realizate.

I. Disciplina: *Practică pentru proiectul de diplomă*

II. Statutul disciplinei: ☒ obligatoriu ☐ opțional ☐ facultativ

III. Precondiții

IV. Conținutul disciplinei:

1. Alegerea temei
2. Căutarea informației de specialitate în țară și străinătate privind tema aleasă.
3. Sinteza stadiului actual în domeniul temei
4. Stabilirea obiectivelor de realizat
5. Stabilirea variantei de implementare, hardware/software
6. Realizare caiet practică

V. Bibliografia minimală obligatorie

- Bibliografia va fi stabilită împreună cu îndrumătorul de diplomă pentru fiecare lucrare în parte.

VI. Fond de timp alocat pe forme de activitate

Semestrul	Forme de activitate/ număr de ore				Număr de credite
	Curs	Seminar	Laborator	Proiect	
6					4

VII. Procedura de evaluare a cunoștințelor: *colocviu*