

PROGRAMAREA INDEPENDENTĂ DE PLATFORMĂ
1. Date despre disciplină

| | | | | | |
|--|---|------------------------|------------------------------|-----------------------------------|---------------------|
| Facultatea | Calculatoare, Informatică și Microelectronică | | | | |
| Departamentul | Ingineria Software și Automatică | | | | |
| Ciclul de studii | Studii superioare de licență, ciclul I | | | | |
| Programul de studii | Automatică și Informatică | | | | |
| Anul de studii | Semestrul | Tip de evaluare | Categoria formativă | Categoria de opționalitate | Credite ECTS |
| Anul IV (<i>învățământ cu frecvență</i>) | 7 | E | S-Disciplina 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 | Practice | Studiul materialului teoretic | Pregătire aplicații |
| Învățământ cu frecvență | 120 | 30 | 30 | 30 | 30 |

3. Precondiții de acces la disciplină

| | |
|---------------------------------------|---|
| Conform planului de învățământ | Matematica, Fizica, Programarea Calculatoarelor, SDA, Programarea Orientată pe Obiect. Legături interdisciplinare cu disciplinele studiate concomitent: Analiza matematică, Programarea calculatoarelor, Grafica pe calculator, Sisteme cu microprocesoare, Arhitecturi de calculatoare, Sisteme de operare |
| Conform competențelor | Aplicarea limbajelor de programare, a mediilor de modelare și dezvoltare, a metodologiilor pentru crearea de programe software |

4. Condiții de desfășurare a procesului educațional pentru

| | |
|-------------------------|--|
| Curs | Pentru prezentarea materialului teoretic în sala de curs este nevoie de proiector și calculator. |
| Lucrări practice | Studentii vor perfecta rapoarte conform condițiilor impuse de indicațiile metodice. Termenul de predare a lucrării este o săptămână după finalizarea acesteia. |

5. Competențe specifice acumulate

| | |
|--------------------------------|---|
| Competențe profesionale | <p>CP1. Utilizarea de cunoștințe de matematică, fizică, tehnica măsurării, grafică inginerescă, mecanică, electrică și electronică, în ingineria sistemelor. Utilizarea în comunicarea profesională a conceptelor, teoriilor și metodelor științelor fundamentale folosite în ingineria sistemelor.</p> <p>Explicarea temelor de rezolvat și argumentarea soluțiilor din ingineria sistemelor, prin utilizarea tehnicilor, conceptelor și principiilor din matematică, fizică, grafică inginerescă, inginerie electrică, electronică.</p> <p>Rezolvarea problemelor uzuale din domeniul ingineriei sistemelor prin identificarea de tehnici, principii, metode adecvate și prin aplicarea matematicii, cu accent pe metodele de calcul numeric.</p> <p>CP3. Utilizarea fundamentelor automatizării, a metodelor de modelare, simulare, identificare și analiză a proceselor, a tehnicilor de proiectare asistată de calculator.</p> <p>Identificarea conceptelor fundamentale ale teoriei sistemelor, ingineriei reglării automate, a principiilor de bază din modelare și simulare, precum și a metodelor de analiză a proceselor, în scopul explicării problemelor de bază din domeniu.</p> <p>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.</p> <p>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.</p> <p>CP4. Proiectarea, implementarea, testarea, utilizarea și mentenanța sistemelor cu echipamente de uz general și dedicat, inclusiv rețele de calculatoare, pentru aplicații de automatică și informatică aplicată.</p> <p>Definirea cu ajutorul principiilor de funcționare și proiectare, a cerințelor standardelor aplicabile și</p> |
|--------------------------------|---|

| | |
|--------------------------------|---|
| | a metodelor de implementare, testare, mentenanță și exploatare a echipamentelor folosite în aplicațiile de automată și informatică aplicată. |
| Competențe transversale | <p>CT1. Aplicarea, în contextul respectării legislației, a drepturilor de proprietate intelectuală (inclusiv transfer tehnologic), a metodologiei de certificare a produselor, a principiilor, normelor și valorilor codului de etică profesională în cadrul propriei strategii de muncă riguroasă, eficientă și responsabilă.</p> <p>CT2. Identificarea rolurilor și responsabilităților într-o echipă plurispecializată, luarea deciziilor și atribuirea de sarcini, cu aplicarea de tehnici de relaționare și muncă eficientă în cadrul echipei.</p> <p>CT3. Identificarea oportunităților de formare continuă și valorificarea eficientă a resurselor și tehnicilor de învățare pentru propria dezvoltare.</p> |

6. Obiectivele disciplinei

| | |
|------------------------------|---|
| Obiectivul general | Cursul își propune să ofere studenților cunoștințe cum să dezvolte aplicații software care să funcționeze pe mai multe platforme hardware și sisteme de operare, fără a fi nevoie de modificări ale codului sursă. |
| Obiectivele specifice | <ol style="list-style-type: none"> 1. Cunoașterea principiilor de abstractizare. Studenții vor învăța cum să izoleze codul dependent de platformă și să utilizeze interfețe și biblioteci care să permită o portabilitate ridicată. 2. Folosirea tehnologiilor cross-platform. Familiarizarea cu limbaje de programare și framework-uri care suportă programarea independentă de platformă, cum ar fi Java (JVM), Python, C++ (cu librării cum ar fi Qt), sau alte soluții moderne, precum Electron sau Flutter. 3. Gestionarea compatibilității între platforme. Testarea de aplicații pe diverse sisteme de operare, având grijă de aspecte precum diferitele arhitecturi hardware sau modurile de interacțiune cu utilizatorul. 4. Dezvoltarea abilităților de gândire adaptive. Abordarea unor provocări specifice legate de compatibilitatea codului și gestionarea bibliotecilor și resurselor externe, astfel încât programul să fie cât mai flexibil și robust. |

7. Conținutul disciplinei

| Tematica activităților didactice | Numărul de ore |
|--|-------------------------|
| | învățământ cu frecvență |
| Tematica cursurilor | |
| T1. Introducere în programarea independentă de platformă. | 2 |
| T2. Structura lexicală a limbajului Java. | 2 |
| T3. Tipuri de date. Controlul execuției. Vectori. | 2 |
| T4. Obiecte și clase în Java. Organizarea claselor. | 2 |
| T5. Interfața grafică cu utilizatorul. | 6 |
| T6. Fire de execuție | 4 |
| T7. Programare în rețea | 4 |
| T8. Lucrul cu baza de date. | 4 |
| T9. Lucrul dinamic cu clase. | 4 |
| Total curs: | 30 |
| Tematica lucrărilor practice | |
| LP1. Inițiere în limbajul de programare, configurarea mediului de lucru. | 2 |
| LP2. Tipuri de date și controlul execuției. | 2 |
| LP3. Obiecte și clase în Java. | 2 |
| LP4. Interfața grafică cu utilizatorul. | 8 |
| LP5. Aplicație simplă client-server folosind socket-uri. | 8 |
| LP6. Achiziția datelor de la dispozitiv incorporat. | 8 |
| Total lucrări practice: | 30 |

8. Referințe bibliografice

| | |
|---------------------|---|
| Principale | <ol style="list-style-type: none"> FRĂSINARU, C. Curs practic de Java. București, Matrix-Rom, 2005, 223 p. Disponibil: http://thor.info.uaic.ro/~acf/java/Cristian_Frasinaru-Curs_practic_de_Java.pdf. HORSTMANN, C. S., CORNELL, G. Core Java Volume I, Fundamentals, 11th Edition, Prentice Hall, 2018, 843 p. ISBN 978-0-13-516630-7. SCHMITT, C, SIMPSON, K. HTML5 Cookbook - Solutions and Examples for HTML5 Developers. O'Reilly 2012, 260p. ISBN 978-1-449-39679-4. |
| Suplimentare | <ol style="list-style-type: none"> DEITEL, P., DEITEL, H. Java How to Program, 10th Edition, Pearson, 2014, 1248 p. ISBN 978-0-133-80780-6. KURNIAWAN, B. Java 7: A Beginner's Tutorial, Third Edition, Brainy Software, 2012, 716p. ISBN 978-0-980-83961-6. |

9. Evaluare

| Periodică | | Curentă | Studiu individual | Proiect/teză | Examen |
|---|------|---------|-------------------|--------------|--------|
| EP 1 | EP 2 | | | | |
| Învățământ cu frecvență | | | | | |
| 15% | 15% | 15% | 15% | | 40% |
| <p>Standard minim de performanță</p> <p>Prezența și activitatea la prelegeri și lucrări de laborator;</p> <p>Obținerea notei minime de „5” la fiecare dintre atestări și lucrări de laborator;</p> <p>Demonstrarea în lucrarea de examinare finală a cunoașterii proceselor și tehnologiilor de bază aplicate la programarea independentă de platformă.</p> | | | | | |

10. Criterii de evaluare

| Activitate | Componente evaluare | Metodă de evaluare, Criterii de evaluare | Pondere în nota finală a activității | Ponderea în evaluarea disciplinei |
|--------------------------------|------------------------------|--|--------------------------------------|-----------------------------------|
| Învățământ cu frecvență | | | | |
| Evaluare periodică I | Conținut teoretic, teme 1-5 | Studentul primește o sarcină de lucru în care va realiza un program simplificat independent de platformă. | 100% | 15% |
| Evaluare periodică II | Conținut teoretic, teme 6-9 | Studentul primește o sarcină de lucru în care va realiza un program independent de platformă cu interfață grafică. | 100% | 15% |
| Evaluare curentă | Activitatea practică | Discuții în cadrul orelor practice | 50% | 15% |
| | | Dosar completat cu Rapoarte pentru fiecare lucrare practică | 50% | |
| Studiul individual | Cercetare la temă | Referat/Prezentare/discurs public | 100% | 15% |
| Evaluarea finală | Conținut teoretic și practic | La calculator, în baza biletului individual. | 100% | 40% |