**Fişa disciplinei**

**1. Date despre program**

|  |  |
| --- | --- |
| 1.1 Instituţia de învăţământ superior  | **Universitatea Babeș-Bolyai Cluj-Napoca**  |
| 1.2 Facultatea  | **Facultatea de Matematică și Informatică**  |
| 1.3 Departamentul  | **Informatică**  |
| 1.4 Domeniul de studii  | **Informatică**  |
| 1.5 Ciclul de studii  | **Licență**  |
| 1.6 Programul de studiu / Calificarea  | **Informatică (în limba germană)**  |

# 2. Date despre disciplină

|  |  |  |  |
| --- | --- | --- | --- |
| 2.1 Denumirea disciplinei  |  | Metode avansate de programare  |  |
| 2.2 Titularul activităţilor de curs  |  | Lect. dr. Cătălin Rusu  |  |
| 2.3 Titularul activităţilor de seminar  |  | Lect. dr. Cătălin Rusu  |  |
| 2.4 Anul de studiu  | 1  | 2.5 Semestrul  | 1  |  | 2.6. Tipul de evaluare  | E  | 2.7 Regimul disciplinei  | Obligatorie  |
| 2.8. Codul disciplinei | MLG5008 |

# 3. Timpul total estimat (ore pe semestru al activităţilor didactice)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 3.1 Număr de ore pe săptămână  | 6  | Din care: 3.2 curs  | 2  | 3.3 seminar/laborator  | 4  |
| 3.4 Total ore din planul de învăţământ  | 84  | Din care: 3.5 curs  | 28  | 3.6 seminar/laborator  | 56  |
| Distribuţia fondului de timp:  | Std.  |
| Studiul după manual, suport de curs, bibliografie şi notiţe  | 15  |
| Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate şi pe teren  | 10  |
| Pregătire seminarii/laboratoare, teme, referate, portofolii şi eseuri  | 20  |
| Tutoriat  | 10  |
| Examinări  | 10  |
| Alte activităţi: ..................  | 1  |
| 3.7 Total ore studiu individual  | 66  |   |  |
| 3.8 Total ore pe semestru  | 150  |   |  |
| 3.9 Numărul de credite  | 6  |   |  |

**4. Precondiţii** (acolo unde este cazul)

|  |  |  |
| --- | --- | --- |
| 4.1 de curriculum  | *
 | Programare orientată obiect  |
|  | *
 | Structuri de date și algoritmi  |
| 4.2 de competenţe  | *
 | Conceptele de bază ale progrămarii orientate obiect  |
| **5. Condiţii** (acolo unde este cazul)  |  |
| 5.1 De desfăşurare a cursului  |   |  |
| 5.2 De desfăşurare a seminarului/laboratorului  | *
 | Laboratoare pe calculator  |

# 6. Competenţele specifice acumulate

|  |  |
| --- | --- |
| **Competenţe****profesionale**  | C1.1 Descrierea adecvată a paradigmelor de programare şi a mecanismelor de limbaj specifice, precum şi identificarea diferenţei dintre aspectele de ordin semantic şi sintactic. C1.2 Explicarea unor aplicaţii soft existente, pe niveluri de abstractizare (arhitectură, pachete, clase, metode) utilizând in mod adecvat cunoştinţele de bază C1.3 Elaborarea codurilor sursă adecvate şi testarea unitară a unor componente într-un limbaj de programare cunoscut, pe baza unor specificaţii de proiectare date C1.5 Dezvoltarea de unități de program și elaborarea documentațiilor aferente  |
| **Competenţe****transversale**  | CT1 Aplicarea regulilor de muncă organizată şi eficientă, a unor atitudini responsabile faţă de domeniul didactic-ştiinţific, pentru valorificarea creativă a propriului potenţial, cu respectarea principiilor şi a normelor de etică profesională CT3 Utilizarea unor metode şi tehnici eficiente de învăţare, informare, cercetare şi dezvoltare a capacităţilor de valorificare a cunoştinţelor, de adaptare la cerinţele unei societăţi dinamice și de comunicare în limba română și într-o limbă de circulație internațională  |

# 7. Obiectivele disciplinei (reieşind din grila competenţelor acumulate)

|  |  |  |
| --- | --- | --- |
| 7.1 Obiectivul general al disciplinei  | *
 | Intelegerea si folosirea conceptelor programarii orientate obiect la dezvoltarea unor aplicatii soft de complexitate medie  |
|  | *
 | Intelegerea si folosirea unor sabloane de proiectare pentru dezvoltarea aplicatiilor.  |
|  | *
 | Deprinderea folosirii unor frameworkuri pentru dezvoltarea aplicatiilor soft.  |
| 7.2 Obiectivele specifice  | *
 | Consolidarea deprinderilor de analiza si proiectare a sistemelor soft prin invatarea unui proces liniar simplu de analiza si proiectare orientata obiect.  |
|  | *
 | Insusirea unor elemente de baza ale limbajului de modelare UML.  |
|  | *
 | Insusirea unor sabloane de proiectare care se folosesc pentru structurarea logica a aplicatiilor.  |
|  | *
 | Invatarea limbajului Java.  |
|  |  | Invatarea limbajului C#  |

# 8. Conţinuturi

|  |  |  |
| --- | --- | --- |
| 8.1. Curs  | Metode de Predare  | Observaţii  |
| 1.Introducere in Java I  | Expuneri: concepte, exemple, studii de caz  |   |
| 2. Introducere in Java II  | Expuneri: concepte, exemple, studii de caz  |   |
| 3. IO. Colectii. Tipuri Generice  | Expuneri: concepte, exemple, studii de caz  |   |
| 4. Programare functionala in JAVA. JAVA8. Streams  | Expuneri: concepte, exemple, studii de caz  |   |
| 5. GUI. JAVA FX  | Expuneri: concepte, exemple, studii de caz  |   |

|  |  |  |
| --- | --- | --- |
| 6. Threaduri  | Expuneri: concepte, exemple, studii de caz  |   |
| 7-8. XML: schema, documente. JSON  | Expuneri: concepte, exemple, studii de caz  |   |
| 9. GUI (Teil 2): FXML, CSS. Metaprogramare: reflection, serialization  | Expuneri: concepte, exemple, studii de caz  |   |
| 10. Programare web. JSP  | Expuneri: concepte, exemple, studii de caz  |   |
| 11. Introducere in C#  | Expuneri: concepte, exemple, studii de caz  |   |
| 12. Modele de dezvoltare software  | Expuneri: concepte, exemple, studii de caz  |   |
| 13-14. Sabloane GRASP/SOLID und GoF. Design orientat obiect. UML  | Expuneri: concepte, exemple, studii de caz  |   |
| Bibliografie  1. James Gosling, Bill Joy, Guy Steele, Gilad Bracha, Alex Buckley. The Java™ Language Specification Java SE 7 Edition.
2. E. Gamma, R. Helm, R. Johnson, J. Vlissides, Design Patterns – Elements of Reusable Object Oriented Software, Ed. Addison Wesley, 1994
3. Joseph Albahari and Ben Albahari, C# 4.0 in a Nutshell, Fourth Edition, O’Reilley, 2010
4. \*\*\*, Microsoft Developer Network, Microsoft Inc.,<http://msdn.microsoft.com/>
5. \*\*\*, The Java Tutorial, 2013.<http://download.oracle.com/javase/tutorial/>

 Bibliografie in limba germana: 1. Gregor Fischer, Jurgen Wolff von Gudenberg, Programmieren in Java, Springer, Berlin, Heidelberg, New York, 2005.
2. Detlef Sesse, Grundkurs Programmieren in Java 1, Hanser Verlag, Muenchen, Wien, 2007.
3. Kamper, S., Grundkurs Programmieren mit Visual Basic, Vieweg Teubner, Wiesbaden, 2009.
4. Kaiser, R., Einführung in C++ und objektorientierte .NET Programmierung, Springer, 2009.
5. Rauber, Th., Runger, G., Parallele Programmierung, Springer, 2007.
 |
| 8.2 Seminar  | Metode de Predare  | Observaţii  |
| S1-S2. Programe simple Java  | Discutii, exemple, studii de caz  |   |
| S3-S4. Colectii/Generice  | Discutii, exemple, studii de caz  |   |
| S5-S6. Strategy/Factory  | Discutii, exemple, studii de caz  |   |
| S7-S8. Concurenta  | Discutii, exemple, studii de caz  |   |
| S9-S10. Java Streams I  | Discutii, exemple, studii de caz  |   |
| S11-S12. Java Streams II  | Discutii, exemple, studii de caz  |   |
| S13-S14. Sabloane de Proiectare  | Discutii, exemple, studii de caz  |   |

Bibliografie în limba germană

1. Gregor Fischer, Jurgen Wolff von Gudenberg, Programmieren in Java, Springer, Berlin, Heidelberg, New York, 2005.
2. Detlef Sesse, Grundkurs Programmieren in Java, Hanser Verlag, Munchen, Wien, 2007.
3. Kamper, S., Grundkurs Programmieren mit Visual Basic, Vieweg Teubner, Wiesbaden, 2009.
4. Kaiser, R., Einführung in C++ und objektorientierte .NET Programmierung, Springer, 2009.
5. Rauber, Th., Runger, G., Parallele Programmierung, Springer, 2007.

Bibliografie generală:

1. Eckel, B. Thinking in Java, 4th edition, Prentice Hall, 2006
2. Joseph Albahari, Ben Albahari, C# 4.0 in a Nutshell, 4th edition, O’Reilley, 2010
3. \*\*\*, Microsoft Developer Network, Microsoft Inc., <http://msdn.microsoft.com/>
4. \*\*\*, The Java Tutorial, 2013, <http://download.oracle.com/javase/tutorial/>

# 9. Coroborarea conţinuturilor disciplinei cu aşteptările reprezentanţilor comunităţii epistemice, asociaţiilor profesionale şi angajatori reprezentativi din domeniul aferent programului

|  |  |
| --- | --- |
|   | Cursul respecta recomandarile curriculei ACM si IEEE pentru studiile de informatica.  |
|   | Cursul apare in planurile de invatamant a celor mai importante universitati din tara si strainatate.  |

**10. Evaluare**

|  |  |  |  |
| --- | --- | --- | --- |
| Tip activitate  | 10.1 Criterii de evaluare  | 10.2 metode de evaluare  | 10.3 Pondere din nota finală  |
| 10.4 Curs  | Cunoasterea conceptelor de baza  | Examen scris  | 40 %  |
| Scrierea unui program  | Examen practic  | 30 %  |
| 10.5 Seminar/laborator  | Programele scrise in timpul semestrului  | Documentatie  | 30 %  |
| 10.6 Standard minim de performanţă  |
| Nota fiecărui exercițiu (laborator) trebuie să fie mai mare de 5. Pentru a promova examenul practic, nota minimă trebuie să fie 5. Pentru a promova examenul scris, nota minimă trebuie să fie 5. Doar nota finală se rotunjește la întreg. Participare obligatorie: Seminar: 75%; Laborator: 90%.  |

#  Data completării Titular de curs Titular de seminar

 15.04.2024 Lect. Dr. Cătălin Rusu Lect. Dr. Cătălin Rusu

 Data avizării în departament Director de departament

 Conf. dr. Adrian Sterca