

A TANTÁRGY ADATLAPJA

1. A képzési program adatai

1.1 Felsőoktatási intézmény	Babeş-Bolyai Tudományegyetem
1.2 Kar	Matematika és Informatika Kar
1.3 Intézet	Magyar Matematika és Informatika Intézet
1.4 Szakterület	Informatika
1.5 Képzési szint	Posztgraduális
1.6 Szak / Képesítés	Posztgraduális informatika és szoftverfejlesztés képzés

2. A tantárgy adatai

2.1 A tantárgy neve	Haladó programozási módszerek és környezetek						
2.2 Az előadásért felelős tanár neve	Dr. Simon Károly						
2.3 A szemináriumért felelős tanár neve	Dr. Simon Károly						
2.4 Tanulmányi év	1	2.5 Félév	2	2.6. Értékelés módja	vizsga	2.7 Tantárgy típusa	Kötelező
2.8 Tantárgy azonosító	MLM5126						

3. Teljes becsült idő (az oktatási tevékenység féléves óraszama)

3.1 Heti óraszám	4	Melyből: 3.2 előadás	2	3.3 szeminárium/labor	2
3.4 Tantervben szereplő össz-óraszám	40	Melyből: 3.5 előadás	20	3.6 szeminárium/labor	20
A tanulmányi idő elosztása					óra
A tankönyv, a jegyzet, a szakirodalom vagy saját jegyzetek tanulmányozása					30
Könyvtárban, elektronikus adatbázisokban vagy terepen való további tájékozódás					15
Szemináriumok / laborok, házi feladatok, portofóliók, referátumok, esszék kidolgozása					30
Egyéni készségfejlesztés (tutorálás)					6
Vizsgák					4
Más tevékenységek:					0
3.7 Egyéni munka össz-óraszama			85		
3.8 A félév össz-óraszama			125		
3.9 Kreditszám			5		

4. Előfeltételek (ha vannak)

4.1 Tantervi	<ul style="list-style-type: none"> • Programozás alapok és algoritmika, Programozás és adatszerkezetek, objektumorientált programozás
4.2 Kompetenciabeli	<ul style="list-style-type: none"> • Középfokú programozási ismeretek egy magas szintű programozási nyelvben

5. Feltételek (ha vannak)

5.1 Az előadás lebonyolításának feltételei	<ul style="list-style-type: none"> • Videoprojektorral és táblával felszerelt előadóterem
5.2 A szeminárium/labor lebonyolításának feltételei	<ul style="list-style-type: none"> • Számítógépekkel felszerelt laborterem, a gépeken a Java programozásra megfelelő szoftverekkel (fejlesztői csomag és környezet)

6. Elsajátítandó jellemző kompetenciák

Szakmai kompetenciák	<p>C1.1. Programozási paradigmák és nyelvspecifikus mehanizmusok kielégítő leírása valamint a szemantika és szintaktika közötti különbségek azonosítása.</p> <p>C1.2. Meglévő szoftver alkalmazás magyarázata különböző absztrakciós szinteken (architektúra, csomagok, osztályok, metódusok), a megszerzett alapismereteket felhasználva.</p> <p>C1.3. Forráskód megfelelő szintű készítése és komponensek unit-tesztelése egy ismert programnyelven, adott feladatspecifikáció alapján.</p> <p>C1.4. Alkalmazások tesztelése adott tesztelési terv alapján</p> <p>C1.5. A programegységek fejlesztése és a kapcsolódó dokumentáció megvalósítása</p>
Transzverzális kompetenciák	<p>CT1. Hatékony és precíz munkavégzés szabályainak alkalmazása, az oktatási és tudományos területtel szembeni felelősségteljes hozzáállás tanúsítása egy adott helyzetben rejlő személyes potenciál optimális és kreatív kiaknázása érdekében a szakmai etikai normák és irányelvek betartása mellett.</p> <p>CT3. Hatékony tanulási, informálódási és kutatási technikák használata valamint tudás felhasználási képességek, a dinamikus társadalom kívánalmaihoz való alkalmazkodóképesség és román valamint egy nemzetközi nyelven történő kommunikációs készségek fejlesztése.</p>

7. A tanágy célkitűzései (az elsajátítandó kompetenciáknak megfelelően)

7.1 A tantárgy általános célkitűzése	<ul style="list-style-type: none"> • Összetettebb szoftverrendszerek megírásához szükséges alapvető módszerek és eszközök bemutatása (Java programozási nyelvben), a platform kapcsolódó csomagjainak megismerése, használatuk elsajátítása
7.2 A tantárgy sajátos célkitűzései	<ul style="list-style-type: none"> • Szoftvertervezéssel kapcsolatos alapfogalmak megismerése • Számítógépes programok megírásához szükséges alapvető eszközök használata • A Java platform megismerése, a biztosított csomagok használatának elsajátítása, a programok fejlesztéséhez, fordításához és futtatásához szükséges eszközök ismerete • Az objektumorientált tervezés alapelveinek és a vonatkozó alapvető tervezési mintáknak a megismerése • A gyakorlati szempontoknak megfelelő programozási stílus elsajátítása

8. A tanágy tartalma

8.1 Előadás	Didaktikai módszerek	Megjegyzések
1. Asztali alkalmazások: UI komponenskészletek (AWT, Swing, FX), eseménykezelés, kapcsolódó tervezési minták	<ul style="list-style-type: none"> • Magyarázat • Párbeszéd • Példák • Interaktív bemutatók 	
2. Felhasználói felületek testreszabása, 2D grafika Java-ban	<ul style="list-style-type: none"> • Magyarázat • Párbeszéd • Példák • Interaktív bemutatók 	
3. Végrehajtási szálak: a Thread osztály és a Runnable interfész, szinkronizáció, wait-notify mechanizmus, a concurrency csomag	<ul style="list-style-type: none"> • Magyarázat • Párbeszéd • Példák • Interaktív bemutatók 	

4. I/O műveletek, adatfolyamok, állományok, szerializáció	<ul style="list-style-type: none"> • Magyarázat • Párbeszéd • Példák • Interaktív bemutatók 	
5. Egyszerű hálózati alkalmazások: a TCP és UDP protokollokra épülő alkalmazások, a java.net csomag, a Socket, ServerSocket, DatagramSocket és URL osztályok	<ul style="list-style-type: none"> • Magyarázat • Párbeszéd • Példák • Interaktív bemutatók 	
6. Többrétegű szoftverarchitektúra, környezeti elemzés és az objektumorientált adatmodell felépítése, az interfészek szerepe a rétegek különválasztásában, kapcsolódó tervezési minták	<ul style="list-style-type: none"> • Magyarázat • Párbeszéd • Példák • Interaktív bemutatók 	
7. Perzisztencia alapok (Java-ban): az Abstract DAO Factory tervezési minta és a JDBC API + erőforrások kezelése	<ul style="list-style-type: none"> • Magyarázat • Párbeszéd • Példák • Interaktív bemutatók 	
8. Kivételkezeléssel kapcsolatos receptek többrétegű architektúrával rendelkező alkalmazások esetében és naplózás	<ul style="list-style-type: none"> • Magyarázat • Párbeszéd • Példák • Interaktív bemutatók 	
9. Konfigurációs paraméterek kezelése, nemzetköziesítés és lokalizáció	<ul style="list-style-type: none"> • Magyarázat • Párbeszéd • Példák • Interaktív bemutatók 	
10. Biztonsággal kapcsolatos alapfogalmak	<ul style="list-style-type: none"> • Magyarázat • Párbeszéd • Példák • Interaktív bemutatók 	

Könyvészet

1. James Gosling, Bill Joy, Guy Steele, Gilad Bracha, Alex Buckley. The Java™ Language Specification Java SE 7 Edition.
2. Eckel, B., Thinking in Java, 4th edition, Prentice Hall, 2006
3. Eckel, B.: Thinking in Patterns with Java, 2004. MindView, Inc
4. E. Gamma, R. Helm, R. Johnson, J. Vlissides, Design Patterns – Elements of Reusable Object Oriented Software, Ed. Addison Wesley, 1994
5. ***, The Java Tutorial, 2013. <http://download.oracle.com/javase/tutorial/>
6. Simon Károly, Kenyerünk Java, Egyetemi Kiadó, 2010

8.2 Szeminárium/labor	Didaktikai módszerek	Megjegyzések
1. Egyszerű grafikus felhasználói felülettel rendelkező asztali alkalmazások létrehozása az AWT/Swing/FX eszközkészletek segítségével	<ul style="list-style-type: none"> • Magyarázat • Párbeszéd • Példák • Interaktív bemutatók 	
2. 2D grafika Java-ban és a felhasználói felületek komponenseinek testreszabása	<ul style="list-style-type: none"> • Magyarázat • Párbeszéd • Példák • Interaktív bemutatók 	
3. Végrehajtási szálak kezelése, szinkronizáció	<ul style="list-style-type: none"> • Magyarázat • Párbeszéd • Példák • Interaktív bemutatók 	
4. Állományok kezelése és szerializáció	<ul style="list-style-type: none"> • Magyarázat 	

	<ul style="list-style-type: none"> • Párbeszéd • Példák • Interaktív bemutatók 	
5. Egyszerű hálózati alkalmazások	<ul style="list-style-type: none"> • Magyarázat • Párbeszéd • Példák • Interaktív bemutatók 	
6. Többrétegű szoftverarchitektúra kialakítása	<ul style="list-style-type: none"> • Magyarázat • Párbeszéd • Példák • Interaktív bemutatók 	
7. Perzisztens adatok kezelése a JDBC API segítségével	<ul style="list-style-type: none"> • Magyarázat • Párbeszéd • Példák • Interaktív bemutatók 	
8. Kivételkezelés és naplózás	<ul style="list-style-type: none"> • Magyarázat • Párbeszéd • Példák • Interaktív bemutatók 	
9. Konfigurációs paraméterek kezelése, nemzetköziesítés és lokalizáció	<ul style="list-style-type: none"> • Magyarázat • Párbeszéd • Példák • Interaktív bemutatók 	
10. Alkalmazások biztonságával kapcsolatos egyszerű példák	<ul style="list-style-type: none"> • Magyarázat • Párbeszéd • Példák • Interaktív bemutatók 	

Könyvészet

1. James Gosling, Bill Joy, Guy Steele, Gilad Bracha, Alex Buckley. The Java™ Language Specification Java SE 7 Edition.
2. Eckel, B., Thinking in Java, 4th edition, Prentice Hall, 2006
3. Eckel, B.: Thinking in Patterns with Java, 2004. MindView, Inc
4. E. Gamma, R. Helm, R. Johnson, J. Vlissides, Design Patterns – Elements of Reusable Object Oriented Software, Ed. Addison Wesley, 1994
5. ***, The Java Tutorial, 2013. <http://download.oracle.com/javase/tutorial/>
6. Simon Károly, Kenyerünk Java, Egyetemi Kiadó, 2010

9. A tantárgy tartalmának összhangba hozása az epiztemikus közösség képviselő, a szakmai egyesületek és a szakterület reprezentatív munkáltatóinak elvárásaival

- A tantárgy követi az IEEE és ACM informatikai kurrikulumát
- A tantárgy része a fontos romániai és külföldi egyetemek oktatási programjának
- A tananyagot a szoftverfejlesztő cégek többsége kiemelten fontosnak tartja a programozói képességek megszerzésének szempontjából

10. Értékelés

Tevékenység típusa	10.1 Értékelési kritériumok	10.2 Értékelési módszerek	10.3 Arány a végső jegyben
10.4 Előadás	Az elsajátított ismeretek, azok felidézésének	Írásbeli vizsga	50%

	helyessége és pontossága. Java programok megtervezésének és implementálásának képessége.		
10.5 Szeminárium/labor	Java programok megírásának, fordításának és futtatásának képessége	Gyakorlati vizsga	50%
10.6 A sikeres vizsga minimális követelményei			
<ul style="list-style-type: none"> Minden részeredménynek átmenő osztályzatnak (min. 5) kell lennie 			

Kitöltés dátuma

Az előadás felelőse

A szeminárium felelőse

Dr. Simon Károly, egyetemi adjunktus

Dr. Simon Károly, egyetemi adjunktus

Az intézeti jóváhagyás dátuma

Intézetvezető

Dr. András Szilárd Károly, egyetemi docens