A TANTÁRGY ADATLAPJA

1. A képzési program adatai

1.1 Felsőoktatási intézmény	Babeș-Bolyai Tudományegyetem
1.2 Kar	Matematika és Informatika
1.3 Intézet	Magyar Matematika és Informatika
1.4 Szakterület	Matematika
1.5 Képzési szint	Alap
1.6 Szak / Képesítés	Matematika

2. A tantárgy adatai

2.1 A tantárgy neve (hu) (en) (ro)	**Objektumorientált programozás** Object-oriented programming Programare orientată obiect						
2.2 Az előadásért felelős tanár neve			Dr. Darvay Zsolt egyetemi docens				
2.3 A szemináriumért felelős tanár neve			Dr. Darvay Zsolt egyetemi docens				
2.4 Tanulmányi év	1	2.5 Félév	2	2.6. Értékelés módja	vizsga	2.7 Tantárgy típusa	kötelező – szaktantárgy
2.8 A tantárgy kódja		**MLM5006**					

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

3.1 Heti óraszám	5	melyből: 3.2 előadás	2	3.3 szeminárium / labor	3
3.4 Tantervben szereplő össz-óraszám	70	melyből: 3.5 előadás	28	3.6 szeminárium / labor	42

A tanulmányi idő elosztása:	óra
A tankönyv, a jegyzet, a szakirodalom vagy saját jegyzetek tanulmányozása	21
Könyvtárban, elektronikus adatbázisokban vagy terepen való további tájékozódás	4
Szemináriumok / laborok, házi feladatok, portofóliók, referátumok, esszék kidolgozása	21
Egyéni készségfejlesztés (tutorálás)	3
Vizsgák	6
Más tevékenységek:	

3.7 Egyéni munka össz-óraszáma	55
3.8 A félév össz-óraszáma	125
3.9 Kreditszám	5

4. Előfeltételek (ha vannak)

4.1 Tantervi	• Nincs.
4.2 Kompetenciabeli	• A C programozási nyelv ismerete.

5. Feltételek (ha vannak)

5.1 Az előadás lebonyolításának feltételei	• Táblával és videoprojektorral felszerelt előadóterem.
5.2 A szeminárium / labor lebonyolításának feltételei	• Számítógépes terem, C++.

6. Elsajátítandó jellemző kompetenciák

Szakmai kompetenciák	C1.1 Fogalmak azonosítása, elméletek leírása és a szaknyelv használata. C1.4. Főbb matematikai problématípusok felismerése és a megoldásukhoz szükséges módszerek, technikák kiválasztása. C2.1 Folyamatok és jelenségek leírására használt alapfogalmak azonosítása. C2.2 Az adatfeldolgozás eredményeinek értelmezése. C2.4 Előzetesen létező adatokat feldolgozó feladatmegoldásból származó adatok összehasonlító elemzése. C2.5 Projektek, dolgozatok kidolgozása és bemutatása az adatelemzés eredményeire vonatkozóan. C3.1 Algoritmusok kidolgozásánál és specifikálásánál használt alapfogalmak azonosítása. C3.2 Adatok értelmezése és az algoritmikusan megoldható feladatok megoldása során a megoldás különböző lépéseinek magyarázata. C3.3 Sajátos technikák és módszerek alkalmazása az algoritmusok tervezése során. C3.4 Adott algoritmus előnyeinek és alkalmazási határainak megállapítása, sajátos fogalmak és szakkifejezések segítségével. C3.5 Feladatok algoritmikus megoldására vonatkozó projektek elkészítése.
Transzverzális kompetenciák	CT1 A szervezett és hatékony munka szabályainak, a didaktikai-tudományos területhez való felelősségteljes hozzáállás alkalmazása a saját potenciál kreatív értékesítéséhez, a szakmai etika alapelveinek és normáinak tiszteletben tartásával. CT2 Interdiszciplináris csoportban szervezett tevékenységek hatékony lebonyolítása és az interperszonális kommunikáció, a különféle csoportokhoz való viszony és együttműködés empatikus képességének fejlesztése. CT3 Hatékony módszerek és technikák használata tanulásra, információszerzésre, kutatásra és a tudásszerzési kapacitások fejlesztésére, egy dinamikus társadalom igényeinek való megfelelésre, román és egy nemzetközi nyelven történő kommunikációra.

7. A tantárgy célkitűzései (az elsajátítandó jellemző kompetenciák alapján)

7.1 A tantárgy általános célkitűzése	• Az objektumorientált programozás alapvető fogalmainak és elveinek elsajátítása.
7.2 A tantárgy sajátos célkitűzései	• A C++ programozási nyelv alapjainak megtanulása. • A hagyományos és az objektumorientált tervezés közötti különbségek felismerése. • Az öröklés, a polimorfizmus, a dinamikus kötés és a sablonok használatának a megértése, újrafelhasználható kód fejlesztése érdekében. • Különböző alkalmazások fejlesztése objektumorientált programozással. • A szabványos könyvtár adatszerkezeteinek és algoritmusainak felhasználása. • Mások által megírt osztályok felhasználása egy adott feladat megoldására.

8. A tantárgy tartalma

8.1 Előadás	Didaktikai módszerek	Megjegyzések
1. A C++ nyelv alapelemei. - Lexikális egységek. Operátorok. Konverzió. - Adattípusok. Változók. Állandók. - Változók láthatósága és élettartama. - Névterek.	Előadás, párbeszéd, problematizálás, tanári magyarázat	
2.Utasítások és C++ függvények. - C++ utasítások. - Függvénydeklaráció és definíció a C++-ban. - Függvények túlterhelése.	Előadás, párbeszéd, problematizálás, tanári magyarázat	

- Inline függvények.		
3. Az alaptípusoknál összetettebb típusok és a dinamikus memóriakiosztás a C++-ban. - Tömb és struktúra. - Mutató és referencia típus. - Memóriaterület lefoglalása és felszabadítása. - Függvényekre hivatkozó mutatók és void függvények.	Előadás, párbeszéd, problematizálás, tanári magyarázat	
4. Moduláris programozás C++-ban. - Fejállományok. Könyvtárak. - Absztrakt adattípusok moduláris megvalósítása. - A void típusra hivatkozó mutatók használata az általánosság érdekében.	Előadás, párbeszéd, problematizálás, tanári magyarázat	
5. Objektumorientált programozás C++-ban. - Osztályok és objektumok. - Egy osztály tagjai. Hozzáférésmódosítók. - Konstruktorok és destruktorok - UML osztálydiagramok (tagok, hozzáférés). - Ismeretség és aggregáció megadása UML diagramban	Előadás, párbeszéd, problematizálás, tanári magyarázat	
6. Operátorok túlterhelése. Statikus tagok. Barát függvények és osztályok.	Előadás, párbeszéd, problematizálás, tanári magyarázat	
7. Öröklés - Származtatott osztályok. Egyszeres öröklés. - Helyettesítési elv. - Metódusok felülírása. - Többszörös öröklés. - Osztályok közti specializálási/általánosítási kapcsolat megadása UML-ben.	Előadás, párbeszéd, problematizálás, tanári magyarázat	
8. Polimorfizmus. - Virtuális tagfüggvények. - Dinamikus kötés. - Virtuális öröklés. - Kód újrafelhasználás (öröklés/összetétel). - Konverzió (upcast/downcast).	Előadás, párbeszéd, problematizálás, tanári magyarázat	
9. Objektumorientált tervezés és interfészre alapozott tervezés. - Absztrakt osztály, interfész. - Az interfész megadása UML –ben. - Egy adatszerkezetekből álló könyvtár objektumorientált tervezése.	Előadás, párbeszéd, problematizálás, tanári magyarázat	
10. Bemeneti/kiviteli műveletek. - Adatfolyamok. Bemeneti/kiviteli osztályhierarchiák. - Formátumozott bemenet/kimenet. Módosítók. - Állománykezelés.	Előadás, párbeszéd, problematizálás, tanári magyarázat	
11. Sablonok. - Függvénysablonok. - Osztálysablonok. - Kód újrafelhasználás.	Előadás, párbeszéd, problematizálás, tanári magyarázat	
12. A szabványos C++ könyvtár (STL) - Tárolók és bejárók. - A szabványos könyvtár algoritmusai.	Előadás, párbeszéd, problematizálás, tanári magyarázat	
13. Kivételkezelés. - A kivétel fogalma.	Előadás, párbeszéd, problematizálás,	

- Kivételek kiváltása és kezelése C++-ban.	tanári magyarázat	
14. A Visual C++ fejlesztési környezet. - A grafikus felhasználói felület elemei.	Előadás, párbeszéd, problematizálás, tanári magyarázat	

Könyvészet

1. Alexandrescu, Programarea modernă in C++. Programare generică si modele de proiectare aplicate, Editura Teora, 2002.(Könyvtár).
2. R. Andonie, I. Garbacea, Algoritmi fundamentali. O perspectiva C++, Editura Libris, Cluj_Napoca, 1995 (Könyvtár).
3. Eckel, Thinking in C++, (second edition), vol. 1 & 2, [www.bruceeckel.com]
4. Milewski, C++ in Action, [http://www.relisoft.com/book/], 1997.
5. L. Negrescu, Limbajul C++, Ed. Albastra, Cluj-Napoca 1996(Könyvtár).
6. Vasile Cioban, Zsolt Darvay, Metode evoluate de programare, UBB-Mate_Info, 1999 (Könyvtár).
7. M. Frenţiu, B. Pârv, Elaborarea programelor. Metode şi tehnici moderne, Ed. Promedia, Cluj-Napoca, 1994. (Könyvtár).

8.2 Szeminárium / Labor	Didaktikai módszerek	Megjegyzések
Sz1. Függvényre hivatkozó mutató. Absztrakt adattípus. Moduláris programozás.	feladat, egyéni munka, párbeszéd	
Sz2. Névtér. Osztály több konstruktorral.	feladat, egyéni munka, párbeszéd	
Sz3. Osztály típusú adattag. Kivételkezelés.	feladat, egyéni munka, párbeszéd	
Sz4. Osztálysablon. Operátor túlterhelés.	feladat, egyéni munka, párbeszéd	
Sz5. Virtuális tagfüggvény. Absztrakt osztály.	feladat, egyéni munka, párbeszéd	
Sz6. Típuskonverzió.	feladat, egyéni munka, párbeszéd	
Sz7. Saját adatszerkezet és hozzá tartozó bejáró megvalósítása.	feladat, egyéni munka, párbeszéd	
L1. Bevezető gyakorlatok a C++ programozási nyelvben.	feladat, egyéni munka, párbeszéd	
L2. Egyszerű C++ feladatok függvényekkel.	feladat, egyéni munka, párbeszéd	
L3. Adatvédelem moduláris programozással a C-ben. Névterek a C++-ban.	feladat, egyéni munka, párbeszéd	
L4. Osztályokra vonatkozó alapvető feladatok. Osztály típusú adattag.	feladat, egyéni munka, párbeszéd	
L5. Feladatok kivételkezeléssel.	feladat, egyéni munka, párbeszéd	
L6. Feladatok osztálysablonokkal és függvénysablonokkal.	feladat, egyéni munka, párbeszéd	
L7. Operátorok túlterhelésére vonatkozó feladatok.	feladat, egyéni munka, párbeszéd	
L8. Adatszerkezetek láncolt megvalósítása. Implicit típuskonverzió.	feladat, egyéni munka, párbeszéd	
L9. Explicit típuskonverzió. Virtuális tagfüggvények.	feladat, egyéni munka, párbeszéd	
L10. Virtuális, illetve absztrakt osztályokkal kapcsolatos feladatok	feladat, egyéni munka, párbeszéd	
L11. Saját adatszerkezet és hozzá kapcsolódó bejáró implementálása.	feladat, egyéni munka, párbeszéd	
L12. Az STL tárolóinak használata.	feladat, egyéni munka, párbeszéd	

L13. Az STL algoritmusainak használata.	feladat, egyéni munka, párbeszéd	
L14. Projektek kiértékelése	feladat, egyéni munka, párbeszéd, gyakorlati projekt	

Könyvészet
1. B. Stroustrup, The C++ Programming Language, Addison Wesley, 1997 http://public.research.att.com/~bs/3rd.html.
2. B. Stroustrup: The C++ Programming Language Special Edition, AT&T, 2000.
3. Peter Müller: Introduction to Object-Oriented Programming Using C++, Globewide Network Academy (GNA), www.gnacademy.org/ August 31, 1997 http://www.zib.de/visual/people/mueller/Course/Tutorial/tutorial.html
4. Mark J. Sebern: STL Iterator Classes, Version 1.2 (1/4/1998), http://www.msoe.edu/eecs/ce/courseinfo/stl/iterator.htm
5. C++ Standard Template Library, http://www.cppreference.com/cppstl.html
6. John Copp: C++ Programming Tutorial, http://cplus.about.com/od/beginnerctutorial/l/blcplustut.htm
7. B. Parv, A. Vancea, Fundamentele limbajelor de programare, Microinformatica, Cluj-Napoca, 1996 (Könyvtár).

9. A tárgy tartalmának összhangba hozása az episztemikus közösségek képviselői, a szakmai egyesületek és a szakterület reprezentatív munkáltatói elvárásaival.

- A tantárgy tartalma összhangban van a fontosabb egyetemeken oktatott objektumorientált programozásra, illetve C++ programozási nyelvre vonatkozó előadásokkal.

10. Értékelés

Tevékenység típusa	10.1 Értékelési kritériumok	10.2 Értékelési módszerek	10.3 Aránya a végső jegyben
10.4 Előadás	Az elméleti anyag ismerete	Elméleti teszt	40%
10.5 Szeminárium / Labor	Szemináriumi tevékenység	Szemináriumi gyakorlatok	10%
	Laboratóriumi tevékenység	Laboratóriumi gyakorlatok és házi feladatok	15%
	Összetett alkalmazások készítése	Projektek	10%
	A gyakorlati anyag elsajátítása	Gyakorlati vizsga	25%

10.6 A teljesítmény minimumkövetelményei

- A C és C++ programozási nyelvek alapjainak ismerete. A két nyelv közötti különbségek felismerése.
- Az objektumorientált programozásra vonatkozó fogalmak ismerete.
- A C++ objektumorientált lehetőségeinek kihasználása egy adott alkalmazás fejlesztésekor.
- Az elméleti teszten és a gyakorlati vizsgán legalább 50%-ot kell elérni.

Kitöltés dátuma	Előadás felelőse	Szeminárium felelőse
	Dr. Darvay Zsolt, egyet. docens	Dr. Darvay Zsolt, egyet. docens
2023. ápr. 15.		

Az intézeti jóváhagyás dátuma	Intézetigazgató
2023. ápr. 15.	Dr. András Szilárd, egyet. docens