

A TANTÁRGY ADATLAPJA

LOGIKAI ÉS FUNKCIONÁLIS PROGRAMOZÁS

Egyetemi tanév: 2025 / 2026

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	Alapképzés
1.6 Szak / Képesítés	Informatika (magyar nyelvű képzés)

2. A tantárgy adatai

2.1 A tantárgy neve (hu) (en) (ro)	Logikai és funkcionális programozás / Programare logică și funcțională / Logic and functional programming						
2.2 Az előadásért felelős tanár neve	Dr. Csató Lehel, egyetemi tanár						
2.3 A szemináriumért felelős tanár neve	Dr. Csató Lehel, egyetemi tanár						
2.4 Tanulmányi év	2	2.5 Félév	3	2.6. Értékelés módja	Koll	2.7 Tantárgy típusa	kötelező - szak
2.8 Tantárgy kódja	MLM5009						

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	56	melyből: 3.5 előadás	28	3.6 szeminárium/labor	28
A tanulmányi idő elosztása:					óra
A tankönyv, a jegyzet, a szakirodalom vagy saját jegyzetek tanulmányozása					26
Könyvtárban, elektronikus adatbázisokban vagy terepen való további tájékozódás					10
Szemináriumok / laborok, házi feladatok, portfóliók, referátumok, esszék kidolgozása					16
Egyéni készségfejlesztés (tutorálás)					14
Vizsgák					3
Más tevékenységek:					-
3.7 Egyéni munka össz-óraszama	69				
3.8 A félév össz-óraszama	125				
3.9 Kreditszám	5				

4. Előfeltételek (ha vannak)

4.1 Tantervi	Nincs
4.2 Kompetenciabeli	Alap-programozási készségek, logikai alapok az elsőéves „Matematikai logika” tárgyból.

5. Feltételek (ha vannak)

5.1 Az előadás lebonyolításának feltételei	• Az előadásokhoz video-projektor szükséges. • A példák kifejtéséhez tábla szükséges.
5.2 A szeminárium / labor lebonyolításának feltételei	• A laboratóriumi órák alatt a diákok a számítógépet, az oktató a táblát használja.

6.1. Elsajátítandó jellemző kompetenciák

Szakmai kompetenciák	• C1.1 A programozási paradigmák és a specifikus nyelvi mechanizmusok megfelelő leírása, valamint a szemantikai és a szintaktikai vonatkozások közötti különbség meghatározása, • C1.3 Megfelelő forráskód fejlesztése egy ismert programozási nyelvben és a komponensek egységes tesztelése adott tervezési specifikáció alapján, • C 4.1 Az informatika alapfogalmainak és alapelveinek, valamint a matematikai elméletek és modellek meghatározása • C 4.2 (Formális) Matematikai és számítógépes modellek értelmezése,
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. • 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.

6.2 Tanulási eredmények

Ismeretek	• A hallgató megérti a Prolog programnyelvben írt programokat, • A hallgató ismeri a Haskell programnyelvet, a curry-zés és a típuslevezetés fogalmait.
Képességek	• A hallgató felismeri a deklaratív programozás elemeit – pl. a „list comprehension”-t. • A hallgató használja a deklaratív/funkcionális programozás paradigmáit.
Felelősség és önállóság	• A hallgató képes önállóan dolgozni a deklaratív nyelvek valamelyikén.

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

7.1 A tantárgy általános célkitűzése	• A tantárgy célja a deklaratív programozási paradigmák bemutatása. • A magas-szintű specifikáció, melyet „kompilálni” tudunk, a deklaratív programozás sajátja. • A funkcionális nyelvek más gondolkodási és programmegoldó sémákat követelnek, célunk ezeknek az elsajátítása.
7.2 A tantárgy sajátos célkitűzései	• Új – deklaratív – programozási nyelvek bemutatása: ◦ A logikai programozás: Prolog; ◦ A funkcionális programozás: Haskell; • Az elsajátított programozási nyelvek és programozási módszerek hasznának és használatának az illusztrálása; • A programozási technikák elemzése, érdekes feladatok deklaratív megoldása; • Programok formális jellemzése a lambda-kalkulus segítségével; • Az automatikus típuslevezetés megismerése és alkalmazása.

8. A tantárgy tartalma

8.1 Előadás	Didaktikai módszerek	Megjegyzések
1. hét. Imperatív és deklaratív nyelvek bemutatása, programozási módszerek ismertetése, a programkódok megértésének fokai. Bevezetés az deklaratív programnyelvek családjába.	tanári magyarázat, rávezetés, interaktív kódírás	
2. hét Logikai programozási alapfogalmak; szabályok és tények; szabad és kötött változók; a dedukció elve és a Prolog következtető rendszere.		
3. hét A cél-kifejezés fogalma Prolog-ban, azok kiértékelése, összetett predikátumok, backtracking, listák Prolog-ban.		
4.hét Mintaillesztés fogalma Prolog-ban, a mintaillesztés szabályai, aritmetikai műveletek, logikai műveletek.		
5. hét A Prolog rendszerben a negáció fogalma, atomok és összetett kifejezések felbontása, összetevése, kiértékelése.		
6. hét Rekurzív adatstruktúrák Prolog-ban. Fogalmak ismételése és megerősítése.		
7. hét A funkcionális paradigma ismertetése, különböző funkcionális programnyelv bemutatása, összehasonlítások, funkcionális jellemzők a gyakran használt programnyelvekben		
8. hét A Haskell nyelv elemei, a típusok fogalma, a Haskell programozói környezet, rekurrens kód írása.		
9. hét Listakezelés Haskell-ben listák ábrázolása, listakonstruktorok.		

10. hét Operátorok Haskell-ben, függvénytípusok, típusok levezetése, lambda-függvények.		
11. hét Bevezető a matematikai programmodellbe és a lambda-kalkulusba, példák a lambda-kalkulusra.		
12. hét Haskell példák a lambda-kalkulusra, elméleti kitekintő.		
13. hét A logikai és a funkcionális programmodellek ismételése, következtetések és összefoglaló		
14. hét Feladatok oldása, Kollokvium, Megoldások megbeszélése.	munkáltatás, teszt-feladatok	
Könyvészet [1]. Serban G., Pop H.F. (2006) Elemente avansate de programare in Lisp si Prolog. Aplicatii in Inteligenta Artificiala, Editura Albastra. [2]. Csató L, Egri E (2010) A Logikai és a Funkcionális Programozás Alapjai, Presa Universitara Clujeana példa-link: https://www.cs.ubbcluj.ro/~csatol/publications/CsatoEgri_Logikai_Funkcionalis.pdf (látogatva: 2020. 10. 08.) [3]. Ásványi Tibor - ELTE - logikai programozás oldalai: Prolog (http://aszt.inf.elte.hu/~asvanyi/pl/jegyzetek , látogatva 2021. május 4-én). [4]. Prolog könyv - letölthető Mike Spivey oldaláról, (http://spivey.oriel.ox.ac.uk/mike/logic/index.html) [5]. (***) Learn Prolog Now! (http://www.coli.uni-saarland.de/~kris/learn-prolog-now/) [6]. Szeredi Péter és Benkő Tamás „Nagyhatékonyságú Logikai Programozás” [7]. Szeredi P, Benkő T (2001) Deklaratív programozás, Számítástudományi és Információelméleti Tanszék (Budapest), Tűzött kötés, 226 oldal. [8]. Allen C, Moronuki J (2016) Haskell programming from first principles (draft). [9]. Reede, C. (1989) Elements of Functional Programming, Addison Wesley. [10]. Field A. (1988) Functional Programming, Addison Wesley, New York. [11]. Horváth Zoltán (ELTE programnyelvek tanszék) Funkcionális programozás előadása. példa-link: https://people.inf.elte.hu/jukraai/fp/fp_ea.pdf (látogatva 2021. 01. 08.) [12]. Graham Hutton (2007) Programming in Haskell, Cambridge University Press. példa-link: http://www.cs.nott.ac.uk/~pszgmh/pih.html#slides (látogatva 2021. 01. 08.) [13]. Miran Lipovaca (2011) Learn you a Haskell for Great Good, No Starch Press, San Francisco. példa-link: http://learnyouahaskell.com/ (látogatva 2021. 01. 08.) [14] Hanák Péter – Deklaratív programozás példa-link: https://dp.iit.bme.hu/ (látogatva 2021. 01. 08.)		
8.2 Szeminárium / Labor	Didaktikai módszerek	Megjegyzések
Szeminárium:		
1. alkalom – Genealógiai adatbázis, lekérdezések, listák kezelése.	munkáltatás, programírás és teszt	
2. alkalom – Minta-alapú szabályok alkotása és alkalmazása. Vágások és hatékony programok írása Prolog-ban.		
3. alkalom – Gyűjtőpredikátumok alkalmazása, adatbázisok konzisztenciájának az ellenőrzése. Kereső algoritmusok.		

4. alkalom – A funkcionális programozás: a logikai programozással történő hasonlóság és eltérések.		
5. alkalom – A funkcionális minták alkalmazása: a FILTER, MAP, FOLDR, UNTIL függvények.		
6. alkalom – A típusok levezetése, a Haskell típusosztályai: új típusok kijelentése, típusosztályok instanciálása.		
7. alkalom – Összefoglaló feladatok, haladó témák: monádok és algebrai struktúrák a Haskell-ben.		
Labor		
1. alkalom Ismerkedés a PROLOG-gal, feladatok kitűzése	munkáltatás, individuális feladatok	
2. alkalom Feladatok ellenőrzése, második feladatcsoporthoz bemutatása		
3. alkalom Feladatok ellenőrzése, harmadik feladatcsoporthoz bemutatása		
4. alkalom Feladatok ellenőrzése, Ismerkedés a HASKELL-lel		
5. alkalom Feladatok ellenőrzése, második feladatcsoporthoz bemutatása		
6. alkalom Feladatok ellenőrzése, harmadik feladatcsoporthoz bemutatása		
7. alkalom Feladatok ellenőrzése		
Könyvészet A kurzus részénél megadottal azonos		

9. Az episztemikus közösségek képviselői, a szakmai egyesületek és a szakterület reprezentatív munkáltatói elvárásainak összhangba hozása a tantárgy tartalmával.

- A logikai programozásba – a Prolog nyelven keresztül – történő bevezetés általános gyakorlat az informatika oktatásában (lásd pl. az Amsterdam-i egyetem, a budapesti ELTE és BME egyetemeken történő képzést). - A funkcionális programozásnál használt Haskell programnyelv egyike a legfejlettebb modern funkcionális nyelveknek, melyet kutatásban és oktatásban egyaránt használnak (pl. St. Andrews-i Egyetem Skóciában, a Nijmegen-i Radboud egyetem, a budapesti ELTE, a BME egyetem).

10. Értékelés

Tevékenység típusa	10.1 Értékelési kritériumok	10.2 Értékelési módszerek	10.3 Aránya a végső jegyben
10.4 Előadás	Parciális vizsga a félév végén (vagy bemutató előadás)	Írásbeli vizsga	60% +10%
10.5 Labor és szeminárium	Programozási feladatok bemutatása.	A megoldások pontozása	40%
	Opcionális feladatok	A megoldások pontozása	+10%
10.6 A teljesítmény minimumkövetelményei			
Az elvárt minimális tudás:			
A logikai programozás fogalmainak ismerete, a backtracking alkalmazása, egyszerű Prolog program megírása.			

- A funkcionális programozás alapfogalmainak az ismerete, a típusok ismerete, az absztrakt osztályok ismerete.

Az átmenő jegy feltételei:

- Az évközi tevékenység kötelező a pontjai 60%-ának az összegyűjtése (minimum 24 pont; kizáró feltétel).
- A laborokon történő jelenlét.
- A vizsgán a pontok 60%-ának az összegyűjtése (kizáró jellegű).
- A végső pontszám minimálisan 70 (=5; minimum-követelmény).

11. SDG ikonok (Fenntartható fejlődési célok/ Sustainable Development Goals)



A fenntartható fejlődés általános ikonja

Nem alkalmazható

Kitöltés dátuma

2025.04.10.

Előadás felelőse

Dr. Csató Lehel egyetemi tanár

Labor felelőse

Dr. Csató Lehel egyetemi tanár

Az intézeti jóváhagyás dátuma

2025.04.24.

Intézetigazgató

Dr. András Szilárd egyetemi docens