

A TANTÁRGY ADATLAPJA

1. A képzési program adatai

1.1 Felsőoktatási intézmény	Babeş–Bolyai Tudományegyetem
1.2 Kar	Matematika és Informatika
1.3 Intézet	Magyar Matematika és Informatika
1.4 Szakterület	informatika
1.5 Képzési szint	alap
1.6 Szak / Képesítés	Informatikai-matematika

2. A tantárgy adatai

2.1 A tantárgy neve	A programozás alapjai						
2.2 Az előadásért felelős tanár neve	Ionescu Klára						
2.3 A szemináriumért felelős tanár neve	Ionescu Klára						
2.4 Tanulmányi év	1	2.5 Félév	1	2.6Értékelés módja	kollokv	2.7 Tantárgy típusa	kötelező – kiegészítő

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

3.1 Heti óraszám	6	melyből: 3.2 előadás	2	3.3 szeminárium/labor	4
3.4 Tantervben szereplő össz-óraszám	84	melyből: 3.5 előadás	28	3.6 szeminárium/labor	56
A tanulmányi idő elosztása:					óra
A tankönyv, a jegyzet, a szakirodalom vagy saját jegyzetek tanulmányozása					40
Könyvtárban, elektronikus adatbázisokban vagy terepen való további tájékozódás					5
Szemináriumok / laborok, házi feladatok, portfóliók, referátumok, esszék kidolgozása					40
Egyéni készségfejlesztés (tutorálás)					5
Vizsgák					6
Más tevékenységek:					
3.7 Egyéni munka össz-óraszama					95
3.8 A félév össz-óraszama					150
3.9 Kreditszám					6 (M), 6 (MI), 6 (I)

4. Előfeltételek (ha vannak)

4.1 Tantervi	Nincs
4.2 Kompetenciabeli	Feladatok kijelentéseinek megértése

5. Feltételek

5.1 Az előadás lebonyolításának feltételei	Táblával és video projektorral felszerelt előadó
5.2 A szeminárium / labor lebonyolításának feltételei	Számítógépes terem, a gépeken Pascal/C++

6. Elsajátítandó jellemző kompetenciák

Szakmai kompetenciák	<ul style="list-style-type: none"> • Az algoritmus fogalmának megértése, az algoritmusok ábrázolási módozatainak elsajátítása • Az algoritmusok tervezéséhez szükséges készségek kialakítása, a fegyelmezett, logikus és algoritmikus gondolkozás kialakítása • A strukturált programozás, a moduláris programtervezés, valamint a top-down és bottom-up programtervezés alapszabályainak megismerése és elsajátítása • Adott feladatosztályokhoz tartozó feladatok megoldási algoritmusainak és a szükséges adatszerkezeteknek megismerése és elsajátítása: számok, karakterláncok feldolgozása, sorozatok, kétdimenziós tömbök, keresés, összefésülés, rendezés stb. • A megtervezett algoritmusok implementálása egyszerű Pascal/C/C++ programok segítségével • A legfontosabb programozási módszerek (visszalépéses keresés, oszd meg és uralkodj, mohó algoritmusok) elsajátítása és a megfelelő feladat-megoldási készség kialakítása • Helyes, átlátható programozási stílus kialakítása, a dokumentálás alapszabályainak megismerése
Transzverzális kompetenciák	<ul style="list-style-type: none"> • Matematikai problémák megoldása informatikai eszközökkel.

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

7.1 A tantárgy általános célkitűzése	<ul style="list-style-type: none"> • Modellezési, feladatmegoldói, informatikai szövegértési készségek, jártasságok fejlesztése. • Az alkotókészség fejlesztése. • Egyéni munkára nevelés és a csapatszellem kialakítása.
7.2 A tantárgy sajátos célkitűzései	<ul style="list-style-type: none"> • Fegyelmezett, logikus és algoritmikus gondolkozás kialakítása. • Programozási módszerek elsajátítása és gyakorlása. • A szoftvertervezés alapszabályainak megismerése.

8. A tantárgy tartalma

8.1 Előadások	Didaktikai módszerek	Megjegyzések
1. A SZÁMÍTÓGÉPES FELADATMEGOLDÁS LÉPÉSEI 1.1. A programozói tevékenység 1.2. A feladatmegoldás lépései számítógépes környezetben 1.3. Alkalmazások minőségi szempontjai 2. AZ ALGORITMUSOK ÁBRÁZOLÁSA 2.1. Algoritmusok 2.1.1. Az algoritmus fogalma 2.1.2. Az algoritmusok leírásánál használt elemek 2.2. Algoritmusok ábrázolása folyamatábrák és pszeudokód nyelvek segítségével 2.3. A strukturált programozás alapelvei 2.3.1. Lineáris struktúrák 2.3.2. Elágazási struktúrák 2.3.3. Ismétlő struktúrák 2.3.4. Az alapstruktúrák jelölése pszeudokódban	1. Előadás	[3] pp: 13-34

<p>2.4. A feladatok számítógépes megoldásához fűződő általános kérdések</p> <p>2.4.1. Algoritmusok helyessége</p> <p>2.4.2. Az algoritmus végrehajtásához szükséges idő</p> <p>2.4.3. Az algoritmus által feldolgozott adatok számára szükséges memória mérete</p> <p>2.4.4. Algoritmusok egyszerűsége</p> <p>2.4.5. Algoritmusok optimalitása</p> <p>2.4.6. Algoritmusok létezése</p>	2. Előadás	[3] pp: 35-44
<p>3. LÉPÉSEK FINOMÍTÁSA</p> <p>3.1. Bevezetés és megoldott feladatok</p>	3. Előadás	[3] pp: 51-60 [4] pp: 9-35
<p>4. PROGRAMOZÁSI TÉTELEK</p> <p>4.1. Egyszerű programozási tételek (Összeg és szorzat, Döntés, Kiválasztás, Szekvenciális (lineáris) keresés, Megszámlálás, Minimum- és maximumkiválasztás, Kiválogatás)</p>	4. Előadás	[3] pp: 73-85 [9] pp: 1-34
<p>4.2. Összetett programozási tételek (Szétválogatás, Sorozat halmazzá alakítása, Keresztmetszet, Egyesítés, Összefésülés)</p>	5. Előadás	[3] pp: 87-96 [10] pp: 1-41
<p>5. ALPROGRAMOK</p> <p>5.1. Bevezetés</p> <p>5.2. Algoritmusok és programok fejlesztési módozatai</p> <p>5.2.1. A top-down típusú (fentről lefele) programozás</p> <p>5.2.2. A bottom-up (lentől felfele) programozás</p> <p>5.2.3. Moduláris algoritmustervezés</p> <p>5.3. A moduláris programozás alapszabályai</p> <p>5.3.1. Moduláris dekompozíció</p> <p>5.3.2. Moduláris kompozíció</p> <p>5.3.3. Modulok tulajdonságai</p> <p>5.3.4. A modularitás alapelvei</p> <p>5.4. Algoritmusok tesztelése</p> <p>5.4.1. A fekete doboz módszere</p> <p>5.4.2. Az átlátszó doboz módszere</p>	6. Előadás	[3] pp: 111-119 [4] pp: 42-66
<p>6. RENDEZÉSI ALGORITMUSOK</p> <p>6.1. Bevezetés</p> <p>6.2. Összehasonlításos rendezési módszerek</p> <p>6.2.1. Buborékrendezés</p> <p>6.2.2. Egyszerű felcseréléses rendezés</p> <p>6.2.3. Válogatásos rendezés</p> <p>6.2.4. Minimum/maximum kiválasztásra épülő rendezés</p> <p>6.2.5. Beszűrő rendezés</p> <p>6.3. Rendezések lineáris időben</p> <p>6.3.1. Leszámláló rendezés (ládarendezés)</p> <p>6.3.2. Számjegyes rendezés</p>	7. Előadás	[3] pp: 131-140
<p>7. REKURZÍÓ</p> <p>7.1. Bevezetés és megoldott feladatok</p> <p>7.2. Közvetlen rekurzió</p>	8. Előadás	[3] pp: 141-153
<p>8. A VISSZALÉPÉSES KERESÉS MÓDSZERE (BACKTRACKING)</p> <p>8.1. Bevezetés</p> <p>8.2. A visszalépéses keresés általános bemutatása</p> <p>8.2.1. Iteratív algoritmus</p> <p>8.2.2. Rekurzív algoritmus</p>	9. Előadás	[3] pp: 161-174
<p>8.3. A visszalépéses keresés bővítése</p> <p>8.4. Visszalépéses keresés a síkban</p>	10. Előadás	[3] pp: 175-184

9. AZ OSZD MEG ÉS URALKODJ MÓDSZER (DIVIDE ET IMPERA) 9.1. Bevezetés és megoldott feladatok 9.2. Az oszd meg és uralkodj módszer általános bemutatása	11. és 12. Előadás	[3] pp: 191-209
10. MOHÓ ALGORITMUSOK (GREEDY MÓDSZER) 10.1. Bevezetés 10.2. A mohó algoritmus általános bemutatása	13. Előadás	[3] pp: 213-235
10.3. Heurisztikus mohó algoritmusok	14. Előadás	[3] pp: 236-240
Könyvészet 1. Cormen T., Leiserson C., Rivest R., Stein, C. – <i>Új algoritmusok</i> , Sclar, Budapest, 2003. 2. Horowitz E. – <i>Fundamentals of Data Structures in C++</i> , Computer Science Press, 1995. 3. Ionescu K. – <i>Bevezetés az algoritmikába</i> , Egyetemi Könyvkiadó, Kolozsvár, 2007 4. Kása Z. – <i>Algoritmusok tervezése</i> , Stúdium Könyvkiadó, Kolozsvár, 1994. 5. Knuth D. E. – <i>A számítógép-programozás művészete, I, II, III kötet</i> , 1992. 6. Rónyai, L., Ivanyos, G., Szabó, R. – <i>Algoritmusok</i> , Typotex, Budapest, 1999. 7. Wirth N. – <i>Algorithms + Data Structures = Programs</i> , Prentice Hall Inc., 1976. 8. Sedgewick R. – <i>Algorithms in C++</i> , Addison-Wesley, 1992. 9. Szlávi P., Zsakó L. – <i>Elemi programozási tételek</i> , Neumann János Számítógép-tudományi Társaság, Budapest, 2001. 10. Szlávi P., Zsakó L. – <i>Összetett programozási tételek</i> , Neumann János Számítógép-tudományi Társaság, Budapest, 2001.		
8.2 Szeminárium / Labor	Didaktikai módszerek	Megjegyzések
1. Lineáris struktúrák, Elágazási struktúrák, Ismétlődő struktúrák, Böhm és Jacopini tétele	1. Szeminárium	[3] pp:23-35
2. Elemi algoritmusok (Felcserélés, Maximumérték, Legnagyobb, Palindromszám), érdekes algoritmus: Elnökválasztás	2. Szeminárium	[3] pp:45-50
3. Elemi algoritmusok 2 (Eukleidész algoritmus, Prímszámok, Fibonacci-számok, Háromszög, Fordított szám, Törzstényezők, Konverzió, Gyors hatványozás)	3. Szeminárium	[3] pp:51-72
4. Egyszerű programozási tételek (Összeg és szorzat, Döntés, Kiválasztás, Szekvenciális (lineáris) keresés, Megszámolás, Minimum- és maximumkiválasztás, Kiválogatás)	4. Szeminárium	[3] pp:73-85 [3] pp:97-100 [9] pp: 1-34
5. Összetett programozási tételek (Szétválogatás, Sorozat halmazra alakítása, Keresztmetszet, Egyesítés, Összefésülés)	5. Szeminárium	[3] pp:87-96 [3] pp:101-104 [10] pp: 1-41
6. Alprogramok (polinomok, mátrixok, determináns)	6. Szeminárium	[3] pp:114-116 [3] pp:120-130
7. Rendező algoritmusok (Buborékrendezés, Egyszerű felcseréléses rendezés, Válogatásos rendezés, Minimum/maximum kiválasztásra épülő rendezés, Beszúró rendezés, Leszámláló rendezés, Számjegyes rendezés)	7. Szeminárium	[3] pp:131-136
8. Parciális vizsga	8. Szeminárium	
9. Rekurzív algoritmusok (Egy szó betűinek megfordítása (Szavak sorrendjének megfordítása, Faktoriális, Legnagyobb közös osztó, Számjegyösszeg, Descartes-szorzat, k elemű részhalmazok, Konverzió, Fibonacci-sorozat, Minden részhalmaz, Partíciók, Halmazpartíciók, Kamatos kamat)	9. Szeminárium	[3] pp:145-160
10. Visszalépéses kereséssel megoldandó feladatok 1: 8 királynő a sakktáblán, Variációk, Zárójelek, Legrövidebb utak, Játékok, Szűrjektív függvények, S pénzösszeg kifizetése, Összegkifizetés minimum számú bankjeggyel	10. Szeminárium	[3] pp:163-174
11. Visszalépéses kereséssel megoldandó feladatok 2: Visszalépéses keresés a síkban, Labirintus, Fénykép, Legnagyobb méretű tárgyak	11. Szeminárium	[3] pp:175-190

12. Oszd meg és uralkodj módszerrel megoldandó feladatok: Minimumszámolás, Hatványozás, Bináris keresés, Összefésülésen alapuló rendezés, Gyorsrendezés, Hanoi tornyok, Úszómedence	12. Szeminárium	[3] pp:192-211
13. Mohó algoritmusokkal megoldandó feladatok: Összeg, Az átlagos várakozási idő minimalizálása, Buszmegállók, Autó bérbeadása, Hátizsák, Minimális feszítőfák (Kruskal és Prim), Minimális hosszúságú utak (Dijkstra algoritmus)	13. Szeminárium	[3] pp:216-235
14. Heurisztikus mohó algoritmusokkal megoldandó feladatok: Utazóügynök, Gráfszínézés, Összegkifizetés legkevesebb számú bankjeggyel	14. Szeminárium	[3] pp:236-245
Könyvészet 1) Ionescu K. – <i>Bevezetés az algoritmikába</i> , Egyetemi Könyvkiadó, Kolozsvár, 2007 2) Kása Z. – <i>Algoritmusok tervezése</i> , Stúdium Könyvkiadó, Kolozsvár, 1994. 3) Szlávi P., Zsakó L. – <i>Elemi programozási tételek</i> , Neumann János Számítógép-tudományi Társaság, Budapest, 2001. 4) Szlávi P., Zsakó L. – <i>Összetett programozási tételek</i> , Neumann János Számítógép-tudományi Társaság, Budapest, 2001.		

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

- A tantárgy tartalma megegyezik az egyetemi oktatásban a fontosabb egyetemeken oktatott algoritmusok és programozás bevezető tárgy hagyományos tartalmával.
- A tárgy keretében figyelembe vesszük a számítógép használata nyújtotta lehetőségeket a matematikai problémák vizsgálatában.

10. Értékelés

Tevékenység típusa	10.1 Értékelési kritériumok	10.2 Értékelési módszerek	10.3 Aránya a végső jegyben
10.4 Előadás	Alapfogalmak és algoritmusok ismerete	A félév közepén parciális írásbeli és gyakorlati vizsga	33 %
10.5 Szeminárium / Labor	Házi feladatok (helyesség, stílus, dokumentáltság, indentálás, tesztelés) 33%	A vizsgaidőszakban írásbeli és gyakorlati vizsga	33 %
10.6 A teljesítmény minimumkövetelményei			
<ul style="list-style-type: none"> • Az elemi algoritmusok ismerete, a programozási tételek alkalmazása • Egyszerű rekurzív algoritmusok ismerete • Tudjon megoldani feladatokat visszalépéses kereséssel, oszd meg és uralkodj módszerrel, mohó algoritmussal 			

Kitöltés dátuma
2015. május 5.

Előadás felelőse
dr. Ionescu Klára

Szeminárium felelőse
dr. Ionescu Klára

Az intézeti jóváhagyás dátuma

Intézetigazgató,
Dr. Szenkovits Ferenc, egyet. docens

.....