

## 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	Alapképzés
1.6 Szak / Képesítés	Informatika – magyar vonal

### 2. A tantárgy adatai

2.1 A tantárgy neve	Osztott operációs rendszerek						
2.2 Az előadásért felelős tanár neve	dr. Ruff Laura-Ildikó, adjunktus						
2.3 A szemináriumért felelős tanár neve	dr. Jakab Hunor, tanársegéd						
2.4 Tanulmányi év	II.	2.5 Félév	I.	2.6. Értékelés módja	Vizsga	2.7 Tantárgy típusa	Kötelező

### 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					24
Könyvtárban, elektronikus adatbázisokban vagy terepen való további tájékozódás					10
Szemináriumok / laborok, házi feladatok, portofóliók, referátumok, esszék kidolgozása					24
Egyéni készségfejlesztés (tutorálás)					5
Vizsgák					6
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	<ul style="list-style-type: none"> <li>Számítógép architektúrák</li> </ul>
4.2 Kompetenciabeli	<ul style="list-style-type: none"> <li>C/C++ programozási készség, Unix alapú operációs rendszerek ismerete, shell programozás</li> </ul>

### 5. Feltételek (ha vannak)

5.1 Az előadás lebonyolításának feltételei	<ul style="list-style-type: none"> <li>vetítógép</li> </ul>
5.2 A szeminárium / labor lebonyolításának feltételei	<ul style="list-style-type: none"> <li>saját felhasználói fiók a kar linux szerverén</li> </ul>

## 6. Elsajátítandó jellemző kompetenciák

<b>Szakmai kompetenciák</b>	<ul style="list-style-type: none"> <li>• az operációs rendszerekkel kapcsolatos kérdéskörökben való jártasság: fontosabb összetevők, funkcionalitás, tervezés, különböző algoritmusok, alapelvek ismerete</li> <li>• a mai modern operációs rendszerek felépítésével kapcsolatos kérdéskörök ismerete</li> <li>• folyamatok közti kommunikációs lehetőségek ismerete/használata, többszálúsággal kapcsolatos alapelvek ismerete és ezek gyakorlati alkalmazása, szinkronizációs eszközök ismerete/használata, szinkronizációs problémák megoldása</li> </ul>
<b>Transzverzális kompetenciák</b>	<ul style="list-style-type: none"> <li>• mivel az operációs rendszer vezérli gyakorlatilag a különböző felhasználói folyamatok futását, az ennek működésébe való betekintés, a háttérben zajó események mélyebb ismerete segít a hatékonyabb programfejlesztésben.</li> <li>• feladatmegoldó készség, kreatív gondolkodás</li> </ul>

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

7.1 A tantárgy általános célkitűzése	<ul style="list-style-type: none"> <li>• megismerkedni az operációs rendszerek általános elméletével (fontosabb összetevők, funkcionalitás, tervezés, különböző algoritmusok, alapelvek)</li> <li>• megismerni a mai modern operációs rendszerek felépítésével kapcsolatos kérdésköröket</li> </ul>
7.2 A tantárgy sajátos célkitűzései	<ul style="list-style-type: none"> <li>• a folyamatok közti kommunikáció különböző eszközeinek megismerése</li> <li>• a multitasking-ot illetve többszálúságot (multithreading) támogató operációs rendszerekben fellépő problémák megértése, illetve az ezekkel kapcsolatos megoldási stratégiák elsajátítása</li> </ul>

## 8. A tantárgy tartalma

8.1 Előadás	Didaktikai módszerek	Megjegyzések
<b>1. hét</b> <b>Bevezetés</b> <ul style="list-style-type: none"> <li>- Operációs rendszerek fejlődésének története</li> <li>- Op. rendszer kérdéskörök</li> <li>- Modern rendszerekhez kapcsolódó fogalmak</li> </ul>	előadás, vetítés, magyarázat, dialógus, konkrét példákon keresztül történő szemléltetés, grafikus ábrázolás	
<b>2. hét</b> <b>Folyamatok</b> <ul style="list-style-type: none"> <li>- Folyamatok állapotai (jelentésük, állapotmodellek, mi tette szükségessé egy-egy új állapot bevezetését)</li> <li>- Állapotátmenetek</li> <li>- adatsruktúrák, melyekre az op. rendszernek szüksége van a folyamatkezeléshez</li> </ul>	előadás, vetítés, grafikus ábrázolás, magyarázat, problémafelvetés, korábbi ismeretek aktivizálása	

<ul style="list-style-type: none"> <li>- Memória táblázatok</li> <li>- Ki-/bemeneti eszköz táblázatok</li> <li>- Állománytáblázatok</li> <li>- Folyamattáblázatok</li> <li>- Folyamatkezelés <ul style="list-style-type: none"> <li>- Üzem mód váltás illetve folyamatváltás (melyik miben áll, különbség)</li> </ul> </li> </ul>		
<p><b>3-6. hét</b> <b>Versenyhelyzet</b></p> <ul style="list-style-type: none"> <li>- Alapfogalmak (kölcsonös kizárás, kritikus szakasz, holtpont, éheztesítés)</li> <li>- Megoldandó problémák egymással (különböző mértékben) kölcsönhatásban levő folyamatok esetén</li> </ul> <p><b>Kölcsonös kizárás</b></p> <ul style="list-style-type: none"> <li>- Kialakulásához szükséges feltételek</li> <li>- Kölcsonös kizárás megvalósítása <ul style="list-style-type: none"> <li>- Szoftver megoldás (Dekker algoritmus, Peterson algoritmus)</li> <li>- Hardver támogatás (megszakítások letiltása, speciális utasítások -Test and Set, Exchange -)</li> <li>- Operációs rendszer támogatás (szemafor, monitor, üzenetek)</li> <li>- tipikus példafeladatok (borbélyüzlet, termelő/fogyasztó, író/olvasó probléma)</li> </ul> </li> </ul> <p><b>Holtpont</b></p> <ul style="list-style-type: none"> <li>- Fellépése újrafelhasználható/elfogyó erőforrások esetén</li> <li>- Holtpont kialakulásának feltételei</li> <li>- Holtpont kezelés: <ul style="list-style-type: none"> <li>- Holtpont megelőzése - deadlock prevention (a holtpont kialakulásához szükséges 4 feltétel valamelyikének megszüntetése által)</li> <li>- Holtpont kialakulásának kivédése - deadlock avoidance (folyamat kéréseinek megtagadása indulásból, erőforrás-hozzárendelés megtagadása - "bankár" algoritmus -)</li> <li>- Holtpont felismerése és feloldása <ul style="list-style-type: none"> <li>- holtpont felismerésre szolgáló algoritmus</li> <li>- különböző megközelítések a holtpont feloldására</li> </ul> </li> </ul> </li> <li>- tipikus példafeladat (étkező filozófusok problémája)</li> </ul>	<p>előadás, vetítés, konkrét példákon keresztül történő szemléltetés, magyarázat, beszélgetés-vita, fakultatív önálló munka</p>	
<p><b>7. hét</b></p> <p><b>Szinkronizáló eszközök a gyakorlatban</b></p> <ul style="list-style-type: none"> <li>- IPC csomag szinkronizáló eszközei</li> <li>- POSIX szinkronizáló eszközök</li> <li>- Példaprogramok</li> </ul>	<p>gyakorlati példák bemutatása, megbeszélése, dialógus</p>	

<p><b>8. hét</b></p> <p><b>Szálak, SMP architektúra</b></p> <ul style="list-style-type: none"> <li>- Szálak <ul style="list-style-type: none"> <li>- A fogalom tisztázása (különbségek a folyamathoz képest)</li> <li>- Szálak használatának előnyei</li> <li>- Szálak kezelése (felhasználói- illetve kernelszintű szálak)</li> <li>- Szálak és folyamatok egymáshoz való viszonya</li> </ul> </li> <li>- SMP szervezés</li> </ul>	<p>előadás, vetítés, magyarázat, konkrét példákon keresztül történő szemléltetés, grafikus ábrázolás, problémafelvetés</p>	
<p><b>9. hét</b></p> <p><b>Ütemezés</b></p> <ul style="list-style-type: none"> <li>- Hosszútávú ütemezés</li> <li>- Középtávú ütemezés</li> <li>- Rövidtávú ütemezés <ul style="list-style-type: none"> <li>- osztályozási kritériumok</li> <li>- ütemezési algoritmusok <ul style="list-style-type: none"> <li>- FCFS, round robin, SPN, SRT, HRRN, visszacsatolásos</li> <li>- Fair share (elvé)</li> </ul> </li> </ul> </li> </ul>	<p>előadás, vetítés, magyarázat, dialógus, konkrét példákon keresztül történő szemléltetés, grafikus ábrázolás</p>	
<p><b>10-12. hét</b></p> <p><b>Memóriakezelés</b></p> <ul style="list-style-type: none"> <li>- Memóriakezeléssel kapcsolatos követelmények</li> <li>- Memóriakezelési technikák: <ul style="list-style-type: none"> <li>- Fix particionálás <ul style="list-style-type: none"> <li>- azonos/változó méretű partíciók</li> <li>- előnye/hátránya</li> </ul> </li> <li>- Dinamikus particionálás <ul style="list-style-type: none"> <li>- előnye/hátránya</li> <li>- memóriablokk elhelyezése (best-fit, first-fit, next-fit algoritmus alapján)</li> </ul> </li> <li>- „Buddy” rendszer</li> <li>- Egyszerű lapozás</li> <li>- Egyszerű szegmentálás</li> </ul> </li> <li><b>Virtuális memória kezelése</b> <ul style="list-style-type: none"> <li>- virtuális memória használatának előnye, alapfogalmak</li> <li>- lapozást támogató hardver</li> <li>- lapozás</li> <li>- szegmentálás</li> <li>- lapozás + szegmentálás</li> <li>- Operációs rendszer stratégiák <ul style="list-style-type: none"> <li>- betöltési stratégia (fetch policy)</li> <li>- elhelyezés (placement)</li> <li>- lapcsere (replacement) <ul style="list-style-type: none"> <li>- lapcsere algoritmusok (optimális, a legrégebben használt, FIFO, clock)</li> <li>- lap-pufferelés</li> </ul> </li> <li>- rezidens halmaz kezelése</li> <li>- kiírási stratégia (cleaning policy)</li> </ul> </li> </ul> </li> </ul>	<p>vetítés, előadás, magyarázat, problémafelvetés</p>	

<b>13. hét</b> <b>Ki-/bemenet kezelés</b> <ul style="list-style-type: none"> <li>- ki-/bemeneti eszközök osztályozása</li> <li>- ki-/bemeneti technikák (programozott I/O, megszakításvezérelt I/O, DMA)</li> <li>- a ki-/bemenet kezelő funkció fejlődése</li> <li>- DMA (szerkezete, működése, szerepe, különböző konfigurációk)</li> <li>- tervezési szempontok</li> <li>- logikai szerkezet</li> <li>- I/O puffereles</li> <li>- lemez teljesítménye</li> <li>- lemez ütemezése <ul style="list-style-type: none"> <li>- ütemezési algoritmusok (FIFO, LIFO, SSTF, SCAN, CSCAN)</li> </ul> </li> </ul>	előadás, vetítés, magyarázat, grafikus ábrázolás	
<b>14. hét</b> <b>Osztott rendszerek</b> <ul style="list-style-type: none"> <li>- kliens-szerver kapcsolat</li> <li>- osztott üzenetküldés</li> <li>- távoli eljárásívás</li> </ul>	előadás, vetítés, magyarázat, grafikus ábrázolás, dialógus	
Könyvészet 1. William Stallings, Operating Systems: Internals and Design Principles, 7th ed., Prentice Hall, 2011 2. (www.WilliamStallings.com) 3. A Tannenbaum, Modern Operating Systems 3rd Ed., Prentice Hall 2007 4. A. Silberschatz, P. Galvin and G. Gagne, Operating System Concepts, John Wiley & Sons, Inc., 2002 5. BOIAN F, VANCEA A., BOIAN R., BUFNEA D., STERCA A., COBARZAN C., COJOCAR D., Sisteme de operare Ed. Risoprint, 2006		
8.2 Szeminárium / Labor 1. rendszerhívások 2. jelzések (signals) 3-4. osztott memória - szinkronizálás nélkül 5-6. semaforok és osztott memória 7-8. üzenetsorok 9. POSIX/Solaris szálak (threads) - szinkronizálás nélkül 10-11. POSIX/Solaris szálak (threads) – szinkronizálással 12-13. Ütemezés 14. labortevékenység lezárása	Didaktikai módszerek  házi feladatok megbeszélése/ bemutatása/hibajavítás, fakultatív pluszfeladatok	Megjegyzések
Könyvészet Buzogány László, Folyamatok, <a href="http://www.cs.ubbcluj.ro/~laura/so2/dokumentacio/">http://www.cs.ubbcluj.ro/~laura/so2/dokumentacio/</a> Programming in C UNIX System Calls and Subroutines using C, <a href="http://www.cs.cf.ac.uk/Dave/C/">http://www.cs.cf.ac.uk/Dave/C/</a>		

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

- A tantárgy tartalma megegyezik az egyetemi oktatásban a fontosabb egyetemeken oktatott, operációs rendszerek általános elméletével foglalkozó tantárgyak tartalmával.

## 10. Értékelés

Tevékenység típusa	10.1 Értékelési kritériumok	10.2 Értékelési módszerek	10.3 Aránya a végső jegyben
10.4 Előadás	Válasz helyessége	Rövid teszt kurzusonként (T)	25%
	Válasz helyessége (alapfogalmak ismerete, tanult algoritmusok alkalmazásának készsége alapján)	Írásbeli vizsga (V)	20%
10.5 Szeminárium / Labor	Program helyessége, határidő betartása	Labor-házifeladatok (L)	35%
	Labor idején megoldott, helyesen működő program, bittologatók előadás/részvétel	Pluszpontok (fakultatív pluszfeladatok, bittologatók) (PP)	
	helyesen működő program	Laborvizsga (LV)	20%
10.6 A teljesítmény minimumkövetelményei			
<ul style="list-style-type: none"><li>• Laborvizsgán elért 25 pont (konkrétan megadott alapfunkcionalitások implementálása) a lehetséges 40-ből</li><li>• Írásbeli vizsgán elért 20 pont a lehetséges 40-ből</li><li>• Nincs 2-nél több indokolatlan, laboróráról való hiányzás</li><li>• Vizsgajegy <math>\geq 5</math>, ahol <math>Vizsgajegy = \lceil (T+L+PP+V+LV)*0.05 \rceil</math></li></ul>			

Kitöltés dátuma

2013.04.29.

Előadás felelőse

.....

Szeminárium felelőse

.....

Az intézeti jóváhagyás dátuma

2013.04.30.

Intézetigazgató

.....