

Óbudai Egyetem Alba Regia Kar				Természettudományi és Szoftvertchnológiai Intézet	
Tantárgy neve és kódja:Szoftverfejlesztés párhuzamos architektúrákra (ATXSP3HMNF)Kreditérték: 4 Nappali/Levelező tagozat 2025/26 tanév I. félév					
Szakok, melyeken a tárgyat oktatják: Mérnökinformatikus MSc szak					
Tantárgyfelelős oktató:				Oktatók:	Szűcs István
Előtanulmányi feltételek: (kóddal)					
Heti óraszámok:	Előadás: 2	Tantermi gyak.:	Laborgyakorlat: 2	Konzultáció:	
Számonkérés módja (s,v,f):	Évközi jegy				
A tananyag					
Oktatási cél: Az alapvető párhuzamos programozási paradigmák, architektúrák és technológiák ismereteinek elsajátítása.					
Tematika: A tantárgy célja az alapvető párhuzamos programozási paradigmák bevezetése és bemutatása a python programnyelv segítségével. A manapság elkerülhetetlen programozási környezetek megfelelő bemutatásával és a különböző a párhuzamos programozás szempontjából releváns verziók közötti különbségek bemutatása. Az alapvető konstrukciókon túl, a mai relatíve modern képfeldolgozási problémák, klasszifikáció és inverz problémákon keresztül mutatja be a párhuzamos programozásban rejlő lehetőségeket. Mindemellett a mai kurrens mesterséges intelligencia libraryk segítségével a grafikus processzorok GPU programozási lehetőségeit is körüljárja, a megfelelő profiling és debugging képességekkel együtt.					
Témakör					Óraszám
Előadások/Gyakorlatok:					
1. Párhuzamos programozás feladata, alapjai, megközelítése, történelme.[1] Python programozása, bemutatása, alapvető nyelvi konstrukciók és környezet, történelmi áttekintés.[2] Féléves feladat megbeszélése.					2/2
2. Fejlesztő környezet bemutatása, programozási lehetőségek demonstrációja.					2/2
3. Alapvető párhuzamos programozási elemek.					2/2
4. Ismertebb párhuzamos programozási problémák megismertetése.					2/2
5. SZEGMENTÁCIÓ: neuronhálókkal / mély hálókkal					2/2
6. SZEGMENTÁCIÓ: neuronhálókkal / mély hálókkal					2/2
7. REKONSTRUKCIÓ: Inverz problémák bemutatása, rekonstrukciós algoritmusok.					2/2
8. REKONSTRUKCIÓ: Maximum Likelihood Expectation Maximization (MLEM).					2/2
9. Párhuzamosság kihasználása a GPU-n.					2/2
10. Pytorch GPU programozás további optimalizációja adott problémára, debuggolás, profilozás bemutatása					2/2
11. Elosztott problémák megoldása, megsegítése MPI pythonnal					2/2
12. Skálázódás és hatékonyság mérése, esetleges optimalizációja					2/2
13. Féléves feladatok bemutatása					2/2
14. Legújabb kutatási irányvonalak bemutatása, pótlás					2/2
Félévközi követelmények					
AZ ELŐADÁSOK LÁTOGATÁSA KÖTELEZŐ!					
Az évközi jegy megszerzésének feltételei: Csoportos beadandó GitHub repositoryba, folyamatos számonkérés. A beadandó program csoportos bemutatása prezentációval a 13. oktatási héten.					
A félévzáró érdemjegy kialakításának módszere					
Az érdemjegyet a félév során készítendő feladatrészek értékelése , valamint az elkészített prezentáció 70-30 százalékban határozzák meg. Mindkettő legalább elégséges szintű (50%).kell, hogy legyen. E fölött egyenletes lineáris skála szerint kap érdemjegyet a hallgató.					
13. hét					
A pótlás módja:		Egyetemi szabályzat szerint			

Aláírás feltétele:	
A vizsga módja (írásbeli, szóbeli, teszt, stb):	

Irodalom:	
Kötelező:	[1] Henessy - Patterson, Computer Architecture: A Quantitative Approach [2] https://docs.python.org/3/whatsnew/3.12.html ehhez hasonlóan a többi verió
Ajánlott:	A tárgy weblapjára feltöltött anyagok