

Szoftvertervezés és -fejlesztés Intézet			Mintatanterv szerinti 1. félév 2026-27-1			
Tantárgy neve:	Kódja:	Kredit:	Óraszám			
				ea	tg	lab
Szoftvertervezés és -fejlesztés I.	NIXSF1LBNE	6	heti	3	0	3
Tárgyfelelős: Dr. Sergyán Szabolcs			Beosztás: egyetemi docens			
Oktató: Kiss Dániel						
Előtanulmányi feltételek:	-	-				
Számonkérés módja:	Vizsga					
A tananyag						
Oktatási cél:	A hallgatók algoritmikus gondolkodásának fejlesztése, algoritmus-alkotási készség kialakítása, gyakran használt algoritmusok megismerése. Ennek érdekében a hallgatók megismerkednek a strukturált és az objektum-orientált programozás alapelveivel és módszereivel, valamint egy konkrét objektum-orientált programnyelv használatával.					
Tematika:	Algoritmusok felépítése, vezérlési szerkezetek. Az algoritmus leírásának eszközei. Egyszerű programozási tételek: sorozatszámítás, eldöntés, kiválasztás, lineáris keresés, megszámlálás, maximumkiválasztás. Összetett programozási tételek: másolás, kiválogatás, szétválogatás, metszet, egyesítés, összefuttatás. Programozási tételek összeépítése. Az objektum-orientált paradigma elemei: objektum, osztály, osztályok közötti kapcsolatok. Az OOP megvalósítások általános jellemzői: egységbezárás, adatrejtés, öröklés, többalakúság, kód újrafelhasználás. Rendezések: egyszerű cserés, kiválasztásos, buborék, beillesztéses. Tesztelés és hibakeresés. Keresések és programozási tételek rendezett tömbökben. Halmazok reprezentációja és műveletei. Rekurzív algoritmusok, programozási tételek rekurzív megvalósítása. „Oszd meg és uralkodj!” elvű algoritmusok, gyorsrendezés és összefésülő rendezés. Optimalizálási problémák megoldása dinamikus programozás és mohó stratégia alkalmazásával.					

Féléves ütemezés	
Oktatási hét (konzultáció)	Témakör
1.	Előadás: Algoritmusok, algoritmus leírás eszközei, algoritmusok hatékonyságának meghatározása. Labor: Változók és operátorok C#-ban. Típuskonverzió, változó túlcsordulás.
2.	Előadás: Egyszerű programozási tételek. Labor: Vezérlési szerkezetek használata C#-ban. Rövidzár kiértékelés, véletlenszám generálás, hibakeresés a kódban.
3.	Előadás: Összetett programozási tételek. Labor: Tömbök használata C#-ban. Matematikai lehetőségek. Egyszerű programozási tételek implementálása.
4.	Előadás: Rendező algoritmusok (alapok, egyszerű cserés rendezés, minimum-kiválasztásos rendezés, buborékredezés és javítása, beillesztéses rendezés és javítása). Labor: Metódusok használata C#-ban. Érték- és referenciatípusok, cím szerinti paraméterátadás. Többdimenziós tömbök.
5.	Előadás: Programozási tételek összeépítése. Labor: Összetett programozási tételek implementálása és alkalmazásuk. Rendező algoritmusok implementálása.

6.	Előadás: Önálló algoritmikusalkotás gyakorlása néhány konkrét probléma megoldásával. Labor: Karakterek és karakterláncok kezelése C#-ban. Egyszerű rejtjelező eljárások megvalósítása.
7.	Előadás: Objektum-orientált programozás elmélete I. Labor: Első zárthelyi dolgozat.
8.	Előadás: Objektum-orientált programozás elmélete II. Labor: Objektum-orientált programozás alapjai C#-ban: egyszerű osztályok elkészítése és példányosítása.
9.	Előadás: Rekurzív algoritmusok. Programozási tételek rekurzív megvalósítása. Labor: Objektum-orientált programozás alapjai C#-ban: objektumtömbök.
10.	Előadás: Rendezett tömbök, programozási tételek rendezett tömbökön. Halmazok, halmazműveletek megvalósítása. Labor: Objektum-orientált programozás alapjai C#-ban: tulajdonságok használata, egyszerű gyakorló feladatok megoldása.
11.	Előadás: „Oszd meg és uralkodj!” elvű algoritmusok I. Összefésülő rendezés. Labor: Fájl- és időkezelés C#-ban. Összetett gyakorló feladatok megoldása.
12.	Előadás: „Oszd meg és uralkodj!” elvű algoritmusok II. Gyorsrendezés és k-adik legkisebb elem kiválasztása. Labor: Osztályszintű tagok és használatuk. Felsorolástípus (enum). Gyűjtemények bejárásra.
13.	Előadás: Optimalizálási problémák I. Dinamikus programozással megoldható feladatok: kincsek begyűjtése, 0-1 hátizsák feladat, leghosszabb közös részsorozat. Labor: Második zárthelyi dolgozat.
14.	Előadás: Optimalizálási problémák II. Mohó algoritmusok: pénzkifizetési feladat. A dinamikus programozás és a mohó stratégia összehasonlítása. Labor: Zárthelyi dolgozat pótlása (opcionális).
Félévközi követelmények	
Az aláírás megszerzésének feltételei:	A tárgy félévközi követelményeinek teljesítésével aláírás szerzhető. A laborgyakorlatokon való részvétel kötelező. A hallgatók a félév során két gyakorlati zárthelyi dolgozatot írnak. Az aláírás megszerzéséhez mindkét dolgozatot legalább 50%-os eredménnyel szükséges teljesíteni, továbbá a hiányzások száma nem lépheti túl az összes laboralkalom 30%-át.
Zárthelyi dolgozatok	
Oktatási hét	Témakör
7.	Zárthelyi dolgozat az előadáson és laboron ismertetett témakörökből
13.	Zárthelyi dolgozat az előadáson és laboron ismertetett témakörökből
14.	Az egyik zárthelyi dolgozat javítása vagy pótlása (opcionális)
Pótlás módja	
A ZH / aláírás pótlásának módja:	A javító zárthelyin az egyik dolgozat javítható/pótolható. A javító dolgozat eredménye felülírja a korábban megírt dolgozat eredményét. Ha a hallgató mindkét gyakorlati zárthelyi megírását elmulasztotta, vagy nem teljesítette azokat az aláíráshoz minimálisan szükséges 50-50%-os eredménnyel, akkor aláírást csak a vizsgaidőszakban meghirdetett aláíráspótláson szerezhet. Hiányzás nem pótolható másik gyakorlati alkalmon való részvétellel. Az aláíráspótló vizsgán legalább 50%-os

	teljesítmény szükséges az aláírás megszerzéséhez.
Vizsga módja	
A tantárgy teljesítéséhez a hallgatónak a vizsgaidőszakban sikeres vizsgát kell tennie. A vizsga egy beugró és egy szóbeli részből áll. A legalább elégséges érdemjegy megszerzéséhez mind a beugró, mind a szóbeli rész sikeres teljesítése szükséges.	
Vizsgajegy kialakítása	
A kurzus végső érdemjegyének kialakítása a gyakorlati rész eredménye, valamint a vizsgán elért eredmény alapján történik. Aláíráspótláson teljesített aláírás esetén a gyakorlati rész eredményét elégségesként vesszük figyelembe a végső jegy kialakításánál. A gyakorlaton elért eredmény az alábbi határok szerint alakítható érdemjeggyé.	
Az egyes érdemjegyek ponthatárai:	
0 – 49: Elégtelen (1) 50 – 61: Elégséges (2) 62 – 73: Közepes (3) 74 – 85: Jó (4) 86 – 100: Jeles (5)	
Irodalom	
Kötelező:	Dr. Sergyán Szabolcs: Algoritmusok, adatszerkezetek I. (ÓE NIK jegyzet, 2014) Az egyetemi e-learning rendszerben a kurzusnál elérhető videók és jegyzetek
Ajánlott:	Cormen, Leiserson, Rivest, Stein: Új algoritmusok, Scolar Kiadó, 2003 Bradley L. Jones: C# mesteri szinten. Kiskapu Kiadó, 2004 Reiter István: C# programozás lépésről lépésre (Jedlik Oktatási Stúdió Bt., 2018)