



TANULÁSI ÚTMUTATÓ

Programozás alapjai, tanulási útmutató informatikus felsőoktatási szakképzési szakosoknak

Készítette: Békési József

Dávid Balázs

Hajdu László

SZTE JGYPK

Informatika Alkalmazásai Tanszék

Lektorálta: Tóth Attila

Jelen tananyag a Szegedi Tudományegyetemen készült az Európai Unió támogatásával. Projekt azonosító: EFOP-3.4.3-16-2016-00014.

Alprojekt azonosító: AP2 – Komplex képzés- és szolgáltatásfejlesztés

Altéma azonosító: AP2_JGYPK5 Magyar és idegen nyelvű képzések oktatási innovációja az MTMI területen és tanártovábbképzés



TARTALOMJEGYZÉK

BEVEZETÉS	4
A TANTÁRGY LEÍRÁSA	5
1.1 TANTÁRGY TANÍTÁSÁNAK CÉLJA	5
1.2 A TANTÁRGY TANULÁSI EREDMÉNYEI	5
1.3 A TANTÁRGY TANULMÁNYI ELŐFELTÉTELE(I), PÁRHUZAMOSSÁGA(I)	8
1.4 A TANTÁRGY TANANYAGTARTALMA (FŐBB TÉMAKÖRÖK) – TEMATIKUS EGYSÉGEK	8
1.5 A TANANYAGTARTALOM FELDOLGOZÁSÁNAK IDŐTERVE	1
1.6 AZ ADOTT TUDÁSELEMEK ÁTADÁSÁT ILLETVE ELSAJÁTÍTÁSÁT SEGÍTŐ MUNKAFORMÁK	1
1.7 AZ ADOTT TUDÁSELEMEK ÁTADÁSÁT ILLETVE ELSAJÁTÍTÁSÁT SEGÍTŐ	
MUNKAMÓDSZEREK	1
1.8 ÉVKÖZI TANULMÁNYI KÖVETELMÉNYEK	1
1.9 A MEGSZERZETT TUDÁS ÉS KOMPETENCIÁK ELLENŐRZÉSE ÉS ÉRTÉKELÉSE	1
1.10 A TANTÁRGY TANÍTÁSÁNAK-TANULÁSÁNAK TÁRGYI FELTÉTELEI	1
1.11 A TANTÁRGY MINŐSÉGFEJLESZTÉSI MÓDSZEREI ÉS FEJLESZTÉSI POLITIKÁJA	2
2. A TANTÁRGY TEMATIKUS EGYSÉGEI	
2.1 A JAVA NYELV ALAPJAI, A PROGRAMOK FORDÍTÁSA ÉS MŰKÖDÉSE	1
2.1.1 TANULÁSI FELADATOK	1
2.1.2 ÖNELLENŐRZŐ FELADATOK	1
2.1.3 MEGOLDÓKULCS AZ ÖNELLENŐRZŐ FELADATOKHOZ	1
2.2 VÁLTOZÓK TÍPUSOK	2
2.2.1 TANULÁSI FELADATOK	2
2.2.2 ÖNELLENŐRZŐ FELADATOK	1
2.2.3 MEGOLDÓKULCS AZ ÖNELLENŐRZŐ FELADATOKHOZ	1
2.3 KIFEJEZÉSEK ÉS ÉRTÉKADÁS	2
2.3.1 TANULÁSI FELADATOK	2
2.3.2 ÖNELLENŐRZŐ FELADATOK	1
2.3.3 MEGOLDÓKULCS AZ ÖNELLENŐRZŐ FELADATOKHOZ	1
2.4 VEZÉRLÉSI SZERKEZETEK (ITERÁCIÓK)	2
2.4.1 TANULÁSI FELADATOK	2
2.4.2 ÖNELLENŐRZŐ FELADATOK	1
2.4.3 MEGOLDÓKULCS AZ ÖNELLENŐRZŐ FELADATOKHOZ	1
2.5 VEZÉRLÉSI SZERKEZETEK (ITERÁCIÓK)	2
2.5.1 TANULÁSI FELADATOK	2
2.5.2 ÖNELLENŐRZŐ FELADATOK	1
2.5.3 MEGOLDÓKULCS AZ ÖNELLENŐRZŐ FELADATOKHOZ	1
2.6 EGYDIMENZIÓS TÖMBÖK	2
2.6.1 TANULÁSI FELADATOK	2
2.6.2 ÖNELLENŐRZŐ FELADATOK	1
2.6.3 MEGOLDÓKULCS AZ ÖNELLENŐRZŐ FELADATOKHOZ	1
2.7 KÉTDIMENZIÓS TÖMBÖK	2
2.7.1 TANULÁSI FELADATOK	2
2.7.2 ÖNELLENŐRZŐ FELADATOK	1



2.7.3	MEGOLDÓKULCS AZ ÖNELLENŐRZŐ FELADATOKHOZ	1
2.8	DINAMIKUS PROGRAMOZÁS	2
2.8.1	TANULÁSI FELADATOK	2
2.8.2	ÖNELLENŐRZŐ FELADATOK	1
2.8.3	MEGOLDÓKULCS AZ ÖNELLENŐRZŐ FELADATOKHOZ	1
2.9	KERESÉS, KIVÁLASZTÁS, RENDEZÉS	2
2.9.1	TANULÁSI FELADATOK	2
2.9.2	ÖNELLENŐRZŐ FELADATOK	1
2.9.3	MEGOLDÓKULCS AZ ÖNELLENŐRZŐ FELADATOKHOZ	1
2.10	METÓDUSOK	2
2.10.1	TANULÁSI FELADATOK	2
2.10.2	ÖNELLENŐRZŐ FELADATOK	1
2.10.3	MEGOLDÓKULCS AZ ÖNELLENŐRZŐ FELADATOKHOZ	1
2.11	METÓDUSOK LOKÁLIS VÁLTOZÓI	2
2.11.1	TANULÁSI FELADATOK	2
2.11.2	ÖNELLENŐRZŐ FELADATOK	1
2.11.3	MEGOLDÓKULCS AZ ÖNELLENŐRZŐ FELADATOKHOZ	1
2.12	REKURZIÓ ÉS REKURZÍV PROBLÉMÁK	2
2.12.1	TANULÁSI FELADATOK	2
2.12.2	ÖNELLENŐRZŐ FELADATOK	1
2.12.3	MEGOLDÓKULCS AZ ÖNELLENŐRZŐ FELADATOKHOZ	1
2.13	OBJEKTUMORIENTÁLTSAĞ ALAPJAI	2
2.13.1	TANULÁSI FELADATOK	2
2.13.2	ÖNELLENŐRZŐ FELADATOK	1
2.13.3	MEGOLDÓKULCS AZ ÖNELLENŐRZŐ FELADATOKHOZ	1



Bevezetés

A programozás alapjai kurzus az általános, programozáshoz szükséges ismeretanyagot tartalmazza, és a Java nyelven keresztül bevezeti a hallgatókat a programozás alapvető építőelemeinek ismeretébe. A tantárgy első féléves alapozókurzus és a tartalmazott tudásanyag elengedhetetlen a képzés további programozással kapcsolatos tantárgyainak teljesítéséhez. A tartalmazott tudás napjainkban már részét képezi az informatikusokkal szemben támasztott elvárásoknak, ezért minden képzésnek fontos része a kurzus.

Az oktatás során az előadás ismeretanyaga, és a gyakorlatok ismeretanyaga párhuzamosan halad, így a tanult elméleti tudást a hallgatók gyakorlatban is megismerik és elsajátítják. A félév során az elméleti anyagból és a gyakorlati anyagból párhuzamosan vannak számonkérések, és előre kiadott otthoni megoldandó feladatok.

A kurzus ismeretanyaga, és a gyakorlaton vett példaprogramok a félév során megosztásra kerül a hallgatókkal a Coospace-n keresztül, ahol digitálisan megtalálható minden egyes órán vett anyag. A gyakorlatok során tanult példaprogramok ismerete önmagukban nem elegendő a tantárgy teljesítéséhez, ezért ajánlott az órák során a hallgatónak jegyzetelni, illetve otthon gyakorolni, hogy önállóan is képesek legyenek a félév során kapott problémák megoldására.

A tantárgy tanulásához ajánlott Angster Erzsébet: Objektumorientált tervezés és programozás könyve, mely tartalmazza az alapfogalmakat, illetve szakkifejezéseket, valamint a tantárgy főbb témaköreit a hozzájuk tartozó példákkal együtt.



A tantárgy leírása

A tantárgy megnevezése: Programozás alapjai		A tantárgy kódja: INF-KTER4
A tantárgy kredit-értéke:	3	
A tantárgy teljesítési formája:	kollokvium	
A tantárgy típusa:	szeminárium	
A tantárgy jellege:		
A tantárgy oktatásának ajánlott féléve:	1.	
A tantárgy meghirdetésének gyakorisága:	évente	
A tantárgy óraszám: - kontakt: - egyéni:	3 kontakt óra 3 óra egyéni munka	
A tantárgy heti óraszám:	3	
A tantárgy oktatásának nyelve:	magyar	
A tantárgyat meghirdető tanszék/ szakcsoport:	Informatika Alkalmazásai Tanszék	
A tantárgy felelőse és elérhetősége:	Békési József	
A tantárgyelem oktatója és elérhetősége:	bekesi@jgypk.u-szeged.hu	

1.1 Tantárgy tanításának célja

A kurzus a Java nyelven keresztül betekintést nyújt a programozáshoz szükséges legalapvetőbb tudásba és egyfajta problémamegoldó gondolkodásmódot tanít a hallgatóknak. A programozási feladatok megoldását, illetve az ehhez szükséges lépéseket elemi úton lépésről lépésre mutatja be azért, hogy a hallgató később önállóan is képes legyen alapvető problémákat megoldani. A kurzus hallgatói megtanulják a Java nyelv valamint tetszőleges programozási nyelvek elemeit a változóktól kezdve a kifejezéseken, tömbökön, iterációkon át a metódusokig vagy akár az objektumorientált programozás alapjait. Mindemellett a példákön és elméleti ismeretanyagokon keresztül a cél egyfajta problémamegoldó gondolkodás megtanítása a hallgatókkal, amely a programozóvá válás első fontos és elengedhetetlen mérföldköve.

1.2 A tantárgy tanulási eredményei

Azoknak az előírt szakmai kompetenciáknak, kompetencia-elemeknek (tudás, képesség stb., KKK 7. pont) a felsorolása, amelyek kialakításához a tantárgy jellemzően, érdemben hozzájárul:

Tudás	Képesség	Attitűd	Autonómia/felelősség
Ismeri a programozás építőelemeit és alap módszereit.	Képes önállóan kisebb programok megvalósítására és matematikai problémák	Elkötelezett a minőségi munka iránt, olvasható és jól dokumentált kódot ír és megérti, hogy a jó	Tudatában van, hogy az általa létrehozott programokkal szemben milyen követelményeket



EFOP-3.4.3-16-2016-00014



	programozással történő megoldására.	programozó egyik, ha nem a legfontosabb ismérve, hogy jól tud csapatban dolgozni.	támaszt az esetleges megrendelő, valamint felelősséget vállal az általa létrehozott szellemi termékért.
Ismeri a Java nyelv által kínált eszköztárat, valamint a nyelvspecifikus tulajdonságait és az objektumorientáltság alapjait.	Képes komplex rendszerek alapjainak a megtervezésére és kivitelezésére, valamint hozzájuk tartozó hatékony programrészletek létrehozására.	Megérti, hogy a programozónak folyamatosan képeznie kell önmagát és tisztában kell lennie az éppen aktuális technológiákkal és trendekkel.	
Ismeri a programozással kapcsolatos alapfeladatokat és a rájuk vonatkozó megoldási módszereket.			
Ismeri a szoftverfejlesztéshez és teszteléshez szükséges alapvető módszereket, és képes ezek használatára.			





A tantárggyal kialakítandó konkrét tanulási eredmények:

Tudás	Képesség	Attitűd	Autonómia/felelősség
Tisztában van a Java programok működésével, valamint fordításával. A JVM felépítésével és működésével	Képes kialakítani a programozáshoz szükséges hardver és szoftverkörnyezetet.	Érdeklődik az új programozással kapcsolatos technológiák iránt, valamint törekszik ezek megismerésére és a tudásának a karbantartására.	Önállóan hoz létre kisebb alkalmazásokat, és határozza meg a szükséges technológiákat.
Tisztában van a változók, tömbök, kifejezések szerepével, valamint a különböző megoldási módszerek során a használatukkal és rendelkezik az értelmezésükhöz és használatukhoz szükséges képességekkel.	A követelményeknek és a megrendelő igényeinek megfelelő programoz hoz létre.	Elkötelezett a minőségi kódolás és a best practice használata mellett annak érdekében, hogy szép és olvasható kódot hozzon létre.	Felelősséget vállal az általa létrehozott szoftvertermékért, valamint tisztában van az általa létrehozott programmal kapcsolatos felelősségével és annak következményeivel.
Ismeri a metódusok és függvények szerepét és tisztában van a szerepükkel.	Korszerű technológiákat használ a szoftvertermék létrehozásakor, valamint képes a tudásra építkezve új technológiák elsajátítására.	Nyitott más nyelvek megismerésére és törekszik is a velük kapcsolatos tudás megszerzésére.	Betartja a struktúrált kóddal és annak megfelelő és jól átlátható kommentelésével kapcsolatos elvárásokat, és tisztában van a fontosságukkal.
Tisztában van az objektumorientáltság szemléletmódjával és eszközeivel.	Képes a kapott programozási feladatot maradéktalanul elvégezni és minőségi kódot létrehozni.	Szívesen dolgozik csapatban más fejlesztőkkel együtt, és érdekelt a tudásának ilyen módon történő bővítésére.	
Ismeri a Java nyelven írt programok ellenőrzésének és debugolásának a technikáit és rendelkezik az ehhez	Képes használni a rendelkezésre álló technikákat annak érdekében, hogy megfelelő és szépen struktúrált kódot hozzon létre.		



szükséges tudással.			
---------------------	--	--	--

1.3 A tantárgy tanulmányi előfeltétele(i), párhuzamossága(i)

Előfeltétel(ek): nincs

Párhuzamosság(ok): nincs

1.4 A tantárgy tananyagtartalma (főbb témakörök) – tematikus egységek

- Java programozási alapok (a program alkotóelemei, változó, típus, forrásprogram szerkezete, metódushívás, értékadó utasítás, adatbevitel, megjelenítés konzolon)
- Kifejezések, értékadás (kifejezés, operátorok, típuskonverziók, értékadás, kifejezések kiértékelése, feltétel, paraméterátadás)
- Tömbök (általános fogalom, egydimenziós tömb, többdimenziós tömb, átadás paraméterként)
- Szelekciók (egyágú, kétágú, egymásba ágyazott szelekciók, többágú szelekciók)
- Iterációk, előltesztelő, hátultesztelő ciklus.
- Iterációk, léptető ciklus, ciklusok egymásba ágyazása
- Keresés, kiválasztás, rendezés
- Dinamikus programozási alapok
- Metódusok írása (metódus szintaktikája, paraméterátadás, visszatérés)
- Metódusok lokális változói
- Rekurzió alapjai
- Az objektumorientáltság alapjai





EFOP-3.4.3-16-2016-00014



1.5 A tananyagtartalom feldolgozásának időterve

Kontaktóra			Egyéni óra	
Hét	Óra	Tartalom	Óra	Tartalom
1	3	A java nyelv alapjai a programok fordítása és működése.	3	A fordító kipróbálása és az első otthoni Java program elkészítése, és a futtatáshoz szükséges környezet megértése.
2	3	Változók és típusok.	3	Különböző változók megértése és elemzése. Az egyes adattípusok viselkedésének megértése és használata.
3	3	Kifejezések és értékadás	3	A különböző operátorok használatának gyakorlásra, típuskonverziók, és egyszerűbb matematikai műveletek használata programozási környezetben matematikai feladatok megoldásán keresztül.
4	3	Vezérlési szerkezetek (szelekciók)	3	Szelekciók gyakorlása és megértésük, egyágú, kétágú, egymásba ágyazott és többágú szelekciók tesztelése és kipróbálása programozási feladatokon keresztül
5	3	Vezérlési szerkezetek (iterációk)	3	Elöl és hátultesztelő ciklusok megértése ismétlést megkövetelő programozási feladatokon keresztül.
6	3	Egydimenziós tömbök	3	Egydimenziós tömbök struktúrájának megértése, bejárásuk gyakorlásán keresztül
7	3	Kétdimenziós tömbök	3	Kétdimenziós tömbök struktúrájának megértése, bejárásuk gyakorlásán keresztül.
8	3	Dinamikus programozás	3	Dinamikus programozási feladatok megoldása, és a tömbökkel való kapcsolatának megértése gyakorlásán keresztül.
9	3	Keresés kiválasztás rendezés	3	Keresések kiválasztások és rendezések kulcslépéseinek a felismerése és megismerése, a különböző rendezések létrehozása és gyakorlása.



EFOP-3.4.3-16-2016-00014



10	3	Metódusok	3	A metódus elméleti hátterének megismerése és meglévő programok átírása a segítségükkel. Gyakorlófeladatokon keresztül a szintaktika és használat megértése.
11	3	Metódusok lokális változói	3	A metódusok lokális változóinak összehasonlítása a globális változókkal, és megértésük a különbségek felismerésén keresztül.
12	3	Rekurzió és rekurzív problémák	3	Rekurzív problémák megoldása és egy adott probléma kisebb részproblémákra való bontásának a módszerének a gyakorlása példafeladatok megoldásán keresztül.
13	3	Objektumorientáltság alapjai	3	Az objektumorientált paradigma elemeinek megértése, osztályok és objektumok használatának gyakorlása és önálló, a szemléletmódot használó programok készítése.

1.6 Az adott tudáselemek átadását, illetve elsajátítását segítő munkaformák

A kurzus óráin a tudáselemek átadása frontális munkában történik, valamint a gyakorlás egyéni munka keretein belül. Az órán kívül a hallgatók egyéni munkában dolgoznak.

1.7 Az adott tudáselemek átadását, illetve elsajátítását segítő munkamódszerek

Az órákon az oktató az aktuális anyagnak először az elméleti háttérét mutatja be előadás szerűen, majd közösen a hallgatókkal feladatokat old meg frontális munkával. A frontális munka során példaprogramokat illetve közös gyakorlófeladatokat old meg az oktató a hallgatókkal együtt.

Az óra keretein kívül a hallgatók önálló munkával oldják meg a kiadott feladatokat, melyek ellenőrzését, illetve átbeszélését az oktatóval konzultációs alkalmak során tehetik meg.

1.8 Évközi tanulmányi követelmények

A félév során a hallgató 3 alkalommal hiányozhat a kontaktóráról, pótlási lehetőség nincs. A zárthelyi dolgozatokból 10 db-ot kötelező megírni melyekre az előadás során bármikor sor kerülhet. A félév során az anyaggal kapcsolatos otthoni megoldandó feladatok kerülnek kiadásra melyek megoldása nem kötelező, azonban erősen javasolt. Az anyaggal és a feladatokkal kapcsolatban a hallgatónak egész félév során lehetősége van konzultálni a tárgy oktatójával egyéni konzultáció keretében.

1.9 A megszerzett tudás és kompetenciák ellenőrzése és értékelése

A gyakorlat és az előadás szorosan összefügg így az előadáson gyakorlathoz kapcsolódó számonkérések is vannak. A félév során összesen 12 db kisdolgozat kerül megírásra, amiből legalább 10 db megírása kötelező a félév eredményes teljesítéséhez. A kisdolgozatok definíciókat és programértelmezéseket tartalmaznak. A vizsgára bocsátáshoz a megszerzhető pontok legalább 50%-át el kell érni. A dolgozatokra az előadáson bármikor sor kerülhet. A kollokviumhoz kapcsolódó számonkérés a vizsgaidőszakban történik egy 20 kérdéses teszt formájában a teljes anyagból.

A vizsga értékelése:

- helyes válasz: 4p
- helytelen válasz: 1p
- nincs válasz: 0p

A dolgozat összpontszáma: 4*helyes válaszok száma – helytelen válaszok száma+ feladatok száma

Értékelés:

0-39: elégtelen

40-63: elégséges

64-75: közepes

76-87: jó

88-100: jeles

1.10 A tantárgy tanításának-tanulásának tárgyi feltételei

A kurzus oktatásához számítógépes terem szükséges és megfelelő fejlesztői környezet mely tartalmaz egy tetszőleges szövegszerkesztő programot és telepített JDK-t az elkészített Java programok futtatásához.

Ezek telepítésének a bemutatásához ajánlott internetkapcsolat.



A kontaktórán kívüli munkához a hallgatónak ajánlott egy az órán használthoz hasonló fejlesztői környezettel rendelkező számítógép.

1.11 A tantárgy minőségfejlesztési módszerei és fejlesztési politikája

- Az oktató naprakészen tartja a tananyaggal kapcsolatos tudását, valamint tisztában van a trendekkel, és a munkaerőpiacon a végzett hallgatókkal szemben támasztott követelményekkel
- Az oktató figyelembe veszi a hallgatók észrevételeit és az anyaggal kapcsolatos megjegyzéseit, ha kell, módosítja az alkalmazott tanítási módszereket, illetve az anyaggal kapcsolatos tanulási folyamatot.
- Az oktató folyamatosan felülvizsgálja a számonkérési módszereket a hallgatói eredmények alapján és a tudás átadását legjobban segítő feladatokat és problémákat vet fel a számonkérések során.
- Az oktató folyamatosan kapcsolatban áll külső gyakorlati helyekkel és a visszajelzések alapján a kritikus témakörök ismeretanyagának az átadását hatékonyabbá teszi.



2. A tantárgy tematikus egységei

2.1 A Java nyelv alapjai, a programok fordítása és működése

2.1.1 Tanulási feladatok

Tartalom:**A Java nyelv alapjai**

- Java programok működése és fordítása
- A JVM környezet
- Programok futtatása

A tematikus egység tanulási eredményei:

A hallgató ismerje

- a java programok általános keretét,
- a fordításhoz és futtatáshoz szükséges környezetet
- a java virtual machine elemeit és működését

Szükséges eszközök, anyagok:

- A hallgatók felkészüléséhez felhasználható szakirodalom (jegyzet, tankönyv, egyéb források és segédanyagok:

Kötelező:

Ajánlott: Coospacen található órai anyag

- Egyebek:



Tanóra (Kontaktóra) (1 kontaktóra = 45 perc)			Egyéni hallgatói munkaóra óra (1 egyéni hallgatói munkatóra = 60 perc)		
Ren- delke- zésre álló idő- tartam	Tanulási tevékenység	Különleges instrukciók	Ren- delke- zésre álló idő- tartam	Tanulási tevékenység	Különleges instrukciók
3 óra	Ismerje meg a Java programok fordítását, valamint a main függvény keretét a szintaktikáját és a benne található kulcsszavak jelentését!	A JDK telepítése után a konzolról történő fordításhoz a környezeti változókhoz hozzá kell adni a fordító elérési útját!	3 óra	Próbálja ki mi történik, ha módosítja az alap Hello World program egyes elemeit. Nézze meg hogy milyen fájlok jönnek létre a fordítás során és keressen összefüggést JDK-val kapcsolatos elméleti tudással!	A JDK telepítése után a konzolról történő fordításhoz a környezeti változókhoz hozzá kell adni a fordító elérési útját!



2.1.2 Önellenőrző feladatok

Magyarázza el a JDK elemeit és működését

Magyarázza el a fordító használatát és működését!

Magyarázza el a main függvény szerepét és működését!

2.1.3 Megoldókulcs az önellenőrző feladatokhoz

Kifejtette a következő fogalmakat: forráskód, bájtkód . JVM, JIT. (20 pont)

Kifejtette a fordítás során végbemenő folyamatokat és a létrejövő fájlokat. (20 pont)

Kifejtette a javac és java utasításokat és használatukat. (20 pont)

Kifejtette a fordítási folyamathoz szükséges lépéseket. (20 pont)

Képes volt a main függvény kódjának visszaadására és elmagyarázására. (20 pont)

Az önellenőrzés értékelése:

Maximálisan elérhető pontszám: 100 pont. A sikeres teljesítéshez legalább 51%-os (51 pont) teljesítés szükséges.

50 pontig: elégtelen (1)

62 pontig: elégséges (2)

75 pontig: közepes (3)

88 pontig: jó (4)

100 pontig: jeles (5)

Hallgatói teljesítményértékelő lap

Ellenőrizze, hogy elvégezte-e a tematikus egység valamennyi feladatát! Minden kérdésnél tegyen egy X-et a leginkább megfelelő rovatba, tehát értékelje saját maga a feladat végrehajtását. Ha a felsoroltak közül valamelyik feladat teljesítése nem történt meg vagy lehetetlen volt a teljesítése, tegyen X-et a "Nem" oszlopba.

		Nem	Igen
1.	Ismeri a JDK elemeit.		
2.	Ismeri a main függvényt.		
3.	Ismeri a fordító működését és használatát.		



2.2 Változók típusok

2.2.1 Tanulási feladatok

Tartalom

Változók típusai

- Egészértékű, lebegőpontos és egyéb típusú változók
- Alul, illetve túlsordulás és a változók viselkedése
- Primitív típusok
- Referencia típusok

A tematikus egység tanulási eredményei:

A hallgató képes legyen:

- adattárolásra változókon keresztül,
- megkülönböztetni a különböző típusokat,
- felismerni milyen változót érdemes használni egy adott adat eltárolására,
- felismerni az alul és túlsordulással keletkező hibákat,
- felsorolni az egyes változókat értelmezési tartománnyal együtt.

Szükséges eszközök, anyagok:

- A hallgatók felkészüléséhez felhasználható szakirodalom (jegyzet, tankönyv, egyéb források és segédanyagok:

Kötelező:

Ajánlott:

- Coospace-n található órai anyag
- Angster Erzsébet: Objektumorientált tervezés és programozás könyv
- Egyebek:



Tanóra (Kontaktóra) (1 kontaktóra = 45 perc)			Egyéni hallgatói munkaóra óra (1 egyéni hallgatói munkatóra = 60 perc)		
Ren- delke- zésre álló idő- tartam	Tanulási tevékenység	Különleges instrukciók	Ren- delke- zésre álló idő- tartam	Tanulási tevékenység	Különleges instrukciók
2 óra	Ismerje meg a változók használatát és azt, hogy mikor és mire érdemes használni őket! Sorolja fel az primitív és a referencia típusú változókat és ismerje meg a kettő közötti különbségeket!	Figyeljen a különböző típusokra!	3 óra	Tároljon el általa választott témában adatokat egy programban különböző típusú változókon keresztül!	Figyeljen a különböző típusokra!
1 óra	Tanulmányozza az alul és túlsordulás jelenségét és a változók értelmezési tartományát!	Ábrázolja a legnagyobb és legkisebb számot az értelmezési tartományban majd adjon hozzá, illetve vonjon ki belőle 1-et! Figyelje meg mi történik!			



2.2.2 Önellenőrző feladatok

Sorolja fel a primitív típusú változókat!
Sorolja fel a referencia típusú változókat!
Ismeresse a különbséget primitív és referencia típusú változók között!
Adja meg az egyes változók értelmezési tartományát!
Ismeresse és idézze elő az alul és túlsordulás jelenségét

2.2.3 Megoldókulcs az önellenőrző feladatokhoz

Felsorolta az összes primitív típusú változót (byte, short, int, long, float, double, char, boolean). (20 pont)
Felsorolta az összes referencia típusú változót (tömbök, osztályok, interfészek). (20 pont)
Ismerette, hogy a referencia típusú változó valójában egy mutató. (20 pont)
Felsorolta az egyes változótípusokhoz tartozó értelmezési tartományokat. (20 pont)
Ismerette, hogy az alul és túlsordulás során az értelmezési tartomány elejére, illetve végére képződik az érték. (20 pont)

Az önellenőrzés értékelése:

Maximálisan elérhető pontszám: 100 pont. A sikeres teljesítéshez legalább 51%-os (51 pont) teljesítés szükséges.

50 pontig: elégtelen (1)
62 pontig: elégséges (2)
75 pontig: közepes (3)
88 pontig: jó (4)
100 pontig: jeles (5)

Hallgatói teljesítményértékelő lap

Ellenőrizze, hogy elvégezte-e a tematikus egység valamennyi feladatát! Minden kérdésnél tegyen egy X-et a leginkább megfelelő rovatba, tehát értékelje saját maga a feladat végrehajtását. Ha a felsoroltak közül valamelyik feladat teljesítése nem történt meg vagy lehetetlen volt a teljesítése, tegyen X-et a "Nem" oszlopba.

	Nem	Igen
1. Ismeri a különböző változótípusokat.		
2. Ismeri az alul és túlsordulás jelenségét.		
3. Ismeri a különbséget primitív és referencia típusok között.		
4. Ismeri a különböző típusú változók értelmezési tartományát.		

2.3 Kifejezések és értékadás

2.3.1 Tanulási feladatok

Tartalom

Kifejezések

- Különböző operátorok használata
- Egyszerűbb matematikai műveletek elvégzése
- Egy, illetve többoperandusú műveletek
- Típuskonverziók

Értékadás

- A kifejezések kiértékelése
- Műveletek sorrendje

A tematikus egység tanulási eredményei:

A hallgató képes legyen:

- kifejezések létrehozására
- alapvető és bonyolultabb matematikai műveletek programozásban történő elvégzésére
- típuskonverzió használatára
- egy, illetve többoperandusú műveletek elvégzésére

Szükséges eszközök, anyagok:

- A hallgatók felkészüléséhez felhasználható szakirodalom (jegyzet, tankönyv, egyéb források és segédanyagok:

Kötelező:

Ajánlott:

- Coospace-n található órai anyag
- Angster Erzsébet: Objektumorientált tervezés és programozás könyv
- Egyebek:



Tanóra (Kontaktóra) (1 kontaktóra = 45 perc)			Egyéni hallgatói munkaóra óra (1 egyéni hallgatói munkatúra = 60 perc)		
Ren- delke- zésre álló idő- tartam	Tanulási tevékenység	Különleges instrukciók	Ren- delke- zésre álló idő- tartam	Tanulási tevékenység	Különleges instrukciók
3 óra	Ismerje meg a kifejezések és értékadások, valamint a típuskonverziók használatát, a matematikai műveletek programozási környezetben történő elvégzését, a műveletek sorrendjét! Ismerje meg hogy értékadáskor a kifejezés mely oldala értékelődik ki először!	Figyeljen az osztásra akkor amikor az eredménynek lebegőpontosnak kell lennie!	3 óra	Ültessen át általa kiválasztott matematikai műveleteket és képleteket programozási környezetbe!	Figyeljen az osztásra akkor amikor az eredménynek lebegőpontosnak kell lennie!

2.3.2 Önellenőrző feladatok

Ismertesse a kifejezése bal, illetve jobb oldalának a szerepét!

Ismertesse a típuskonverzió hasznát példával!

Alakítson át egy matematikai formulát programozásbéli megfelelőjére!

Ismertesse mi a különbség az egy, illetve többoperandusú műveletek között!

2.3.3 Megoldókulcs az önellenőrző feladatokhoz

Ismertette, hogy egy kifejezés bal oldalába képződik a jobb oldal eredménye (30 pont)

Bemutatta, hogy a típuskonverzióval különböző típusok hogyan konvertálhatók át más típusra. (30 pont)

Képes volt átalakítani egy tetszőleges formulát úgy, hogy az eredmény nem változott. (20 pont)

Ismertette az egy, illetve többoperandusú műveleteket példával együtt (1: inkrementálás, negáció stb. 2: összeadás kivonás szorzás osztás) (20 pont)

Az önellenőrzés értékelése:

Maximálisan elérhető pontszám: 100 pont. A sikeres teljesítéshez legalább 51%-os (51 pont) teljesítés szükséges.

50 pontig: elégtelen (1)

62 pontig: elégséges (2)

75 pontig: közepes (3)

88 pontig: jó (4)

100 pontig: jeles (5)

Hallgatói teljesítményértékelő lap

Ellenőrizze, hogy elvégezte-e a tematikus egység valamennyi feladatát! Minden kérdésnél tegyen egy X-et a leginkább megfelelő rovatba, tehát értékelje saját maga a feladat végrehajtását. Ha a felsoroltak közül valamelyik feladat teljesítése nem történt meg vagy lehetetlen volt a teljesítése, tegyen X-et a "Nem" oszlopba.

		Nem	Igen
1.	Ismeri a kifejezések szintaktikáját és képes használni őket.		
2.	Képes megváltoztatni egy tetszőleges változó típusát.		
3.	Tisztában van egy kifejezés bal, illetve jobb oldalának szerepével.		
4.	Képes használni minden egy és többoperandusú műveletet.		

2.4 Vezérlési szerkezetek (iterációk)

2.4.1 Tanulási feladatok

Tartalom

Vezérlési szerkezetek

- Egyágú szelekciók
- Kétágú szelekciók
- Egymásba ágyazott szelekciók
- Kifejezések igazságértéke és kapcsolatuk a szelekciókkal

A tematikus egység tanulási eredményei:

A hallgató képes legyen:

- egy, többágú valamint egymásba ágyazott szelekciók létrehozására
- a szelekciókhoz kapcsolódó feladatok megoldására
- különböző programágak létrehozására szelekcióval
- szelekciókkal kapcsolatos feltételek megfogalmazására és helyes használatukra

Szükséges eszközök, anyagok:

- A hallgatók felkészüléséhez felhasználható szakirodalom (jegyzet, tankönyv, egyéb források és segédanyagok:

Kötelező:

Ajánlott:

- Coospace-n található órai anyag
- Angster Erzsébet: Objektumorientált tervezés és programozás könyv
- Egyebek:



Tanóra (Kontaktóra) (1 kontaktóra = 45 perc)			Egyéni hallgatói munkaóra óra (1 egyéni hallgatói munkatóra = 60 perc)		
Ren- delke- zésre álló idő- tartam	Tanulási tevékenység	Különleges instrukciók	Ren- delke- zésre álló idő- tartam	Tanulási tevékenység	Különleges instrukciók
2 óra	Ismerje meg a különbséget az egyágú, többágú illetve egymásba ágyazott szelekciók között.	Figyeljen a szelekciók sorrendjére!	2 óra	Hozzon létre önállóan a tanult szelekciók közül mindegyik fajtából, majd értelmezze a létrehozott vezérlési szerkezeteket!	Figyeljen a szelekciók sorrendjére!
1 óra	Ismerje meg a kifejezéseket, valamint ezzel kapcsolatos kérdések megfogalmazása és formalizálásukat feltételekkel	Több feltétel is megadható egyszerre melyek együtt értékelődnek ki!	1 óra	Fogalmazzon meg feltételeket saját maga, melyeket programozási környezetbe ültet át!	Több feltétel is megadható egyszerre melyek együtt értékelődnek ki!

2.4.2 Önellenőrző feladatok

Ismertesse a vezérlési szerkezetek szerepét!

Ismertesse a szelekciók különböző típusait példával együtt!

Ismertesse a szelekciókhoz tartozó kifejezések szintaktikai szabályait!

2.4.3 Megoldókulcs az önellenőrző feladatokhoz

Ismertette, hogy a vezérlési szerkezetekkel a program futását tudjuk módosítani, a szelekciókkal pedig különböző ágakat létrehozni. (30 pont)

Bemutatta a szelekciók különböző típusait az egy, többágú valamint egymásba ágyazott szelekciókat (20 pont)

Kitért a szelekciók esetén az egyes kifejezések és ágak sorrendjének a szerepére (20 pont)

Bemutatta a szelekciókhoz kapcsolódó kifejezéseket és kitért a több kifejezésből álló feltételekre is. (30 pont)

Az önellenőrzés értékelése:

Maximálisan elérhető pontszám: 100 pont. A sikeres teljesítéshez legalább 51%-os (51 pont) teljesítés szükséges.

50 pontig: elégtelen (1)

62 pontig: elégséges (2)

75 pontig: közepes (3)

88 pontig: jó (4)

100 pontig: jeles (5)

Hallgatói teljesítményértékelő lap

Ellenőrizze, hogy elvégezte-e a tematikus egység valamennyi feladatát! Minden kérdésnél tegyen egy X-et a leginkább megfelelő rovatba, tehát értékelje saját maga a feladat végrehajtását. Ha a felsoroltak közül valamelyik feladat teljesítése nem történt meg vagy lehetetlen volt a teljesítése, tegyen X-et a "Nem" oszlopba.

		Nem	Igen
1.	Tisztában van a szelekciók szerepével a programozásban.		
2.	Képes önállóan használni a szelekciók egyes fajtáit.		
3.	Ismeri a kapcsolódó kifejezéseket és használatukat.		

2.5 Vezérlési szerkezetek (iterációk)

2.5.1 Tanulási feladatok

Tartalom

Iterációk

- Elöltesztelő ciklus
- Hátultesztelő ciklus
- Számláló ciklus
- Ciklusváltozó, ciklusmag, break és continue utasítások

A tematikus egység tanulási eredményei:

A hallgató képes legyen:

- az előltesztelő és a hátultesztelő ciklus közötti különbség ismertetésére,
- növekvő és csökkenő számláló, elől és hátultesztelő ciklus használatára,
- a különböző ciklusok megfeleltetésére egymással,
- használni a break és continue utasításokat

Szükséges eszközök, anyagok:

- A hallgatók felkészüléséhez felhasználható szakirodalom (jegyzet, tankönyv, egyéb források és segédanyagok:

Kötelező:

Ajánlott:

- Coospace-n található órai anyag
- Angster Erzsébet: Objektumorientált tervezés és programozás könyv
- Egyebek:



Tanóra (Kontaktóra) (1 kontaktóra = 45 perc)			Egyéni hallgatói munkaóra óra (1 egyéni hallgatói munkatóra = 60 perc)		
Ren- delke- zésre álló idő- tartam	Tanulási tevékenység	Különleges instrukciók	Ren- delke- zésre álló idő- tartam	Tanulási tevékenység	Különleges instrukciók
3 óra	Ismerje meg a While, a Do While és a For ciklus működését, csökkenő, illetve növekvő esetben! Ismerje meg a megfeleltetést a különböző típusú ciklusok között! Ismerje meg a break és continue kulcsszavakat valamint használatukat!	Végtelen ciklusból ki lehet ugrani egy feltétel teljesülésekor egy break utasítással!	3 óra	Írjon végtelen ciklust, ismerje meg a ciklusváltó szerepét és a megállási feltétel fogalmát! Találjon ki példákat melyekre használhatóak az ismétléses vezérlések, és valósítsa meg őket!	Végtelen ciklusból ki lehet ugrani egy feltétel teljesülésekor egy break utasítással!

2.5.2 Önellenőrző feladatok

Magyarázza el az különböző típusú ciklusokat!

Magyarázza el milyen változó játszik kulcsszerepet a ciklusok működésekor!

Magyarázza el a megállási feltétel fogalmát!

2.5.3 Megoldókulcs az önellenőrző feladatokhoz

Kifejtette a következő kulcsszavakat: While, Do While, For (20 pont)

Kifejtette, hogy a Do While egyszer mindenképpen végrehajtódik míg a While nem. (20 pont)

Kifejtette a ciklusváltozó fogalmát, valamint a megállási feltétellel való kapcsolatát (20 pont)

Kifejtette a következő kulcsszavakat: break, continue (20 pont)

Kifejtette, hogy a break utasítással ki tudunk ugrani egy ciklusból, valamint a continue utasítással a következő ciklusmagra tudunk ugrani (20 pont)

Az önellenőrzés értékelése:

Maximálisan elérhető pontszám: 100 pont. A sikeres teljesítéshez legalább 51%-os (51 pont) teljesítés szükséges.

50 pontig: elégtelen (1)

62 pontig: elégséges (2)

75 pontig: közepes (3)

88 pontig: jó (4)

100 pontig: jeles (5)

Hallgatói teljesítményértékelő lap

Ellenőrizze, hogy elvégezte-e a tematikus egység valamennyi feladatát! Minden kérdésnél tegyen egy X-et a leginkább megfelelő rovatba, tehát értékelje saját maga a feladat végrehajtását. Ha a felsoroltak közül valamelyik feladat teljesítése nem történt meg vagy lehetetlen volt a teljesítése, tegyen X-et a "Nem" oszlopba.

	Nem	Igen
1. Ismeri a különböző ciklusfajtákat.		
2. Ismeri a ciklusváltozó és ciklusmag fogalmát és szerepét.		
3. Ismeri a break és a continue utasításokat.		
4. Ismeri a tipikus feladatokat melyekre a ciklusok használhatók.		

2.6 Egydimenziós tömbök

2.6.1 Tanulási feladatok

Tartalom

Egydimenziós tömbök

- Tömbök kapcsolata a ciklusokkal
- Tömbök létrehozása
- Tömbök mérete és bejárása

A tematikus egység tanulási eredményei:

A hallgató képes legyen:

- egydimenziós tömbök elméleti hátterének ismertetésére
- tömböt létrehozni, feltölteni, illetve bejárni valamint az elemeket elérni és módosítani.

Szükséges eszközök, anyagok:

- A hallgatók felkészüléséhez felhasználható szakirodalom (jegyzet, tankönyv, egyéb források és segédanyagok:

Kötelező:

Ajánlott:

- Coospace-n található órai anyag
- Angster Erzsébet: Objektumorientált tervezés és programozás könyv
- Egyebek:



Tanóra (Kontaktóra) (1 kontaktóra = 45 perc)			Egyéni hallgatói munkaóra óra (1 egyéni hallgatói munkatóra = 60 perc)		
Ren- delke- zésre álló idő- tartam	Tanulási tevékenység	Különleges instrukciók	Ren- delke- zésre álló idő- tartam	Tanulási tevékenység	Különleges instrukciók
3 óra	Ismerje meg a tömbök létrehozását, feltöltését, indexelésének módszerét! Ismerje meg tetszőleges méretű tömb bejárását! Ismerje meg a ciklusváltozók szerepét a tömbök bejárása során! Ismerje meg az indexek kapcsolatát a ciklusváltozókkal a bejárás esetén!	Ha a ciklus megállási feltételének a tömb méretét adjuk, akkor tetszőleges méretű tömböt képesek vagyunk bejárni!	3 óra	Soroljon fel gyakorlati példákat, amikor hasznosak lehetnek a tömbök! Hozzon létre sorozatokat az egydimenziós tömbök segítségével és járja be őket!	Figyeljen rá, hogy a tömb indexelése 0-val kezdődik!

2.6.2 Önellenőrző feladatok

Magyarázza el a tömbök létrehozását és felépítését!

Magyarázza el a tömbök indexelésének a tulajdonságait!

Magyarázza el a tömbök bejárását!

Magyarázza el a maximálisan elérhető index meghatározását a tömbben!

2.6.3 Megoldókulcs az önellenőrző feladatokhoz

Kifejtette a következő fogalmakat: tömb mérete, new kulcsszó, szögletes zárójel. (20 pont)

Kifejtette, hogy a tömböket 0-tól kezdődően indexeljük. (20 pont)

Kifejtette a tömbök bejárásának kapcsolatát a ciklusokkal, és ismertette a ciklusváltozó indexként használatát (40 pont)

Kifejtette, hogy a maximálisan elérhető index a tömbméret-1 tetszőleges tömbben (20 pont)

Az önellenőrzés értékelése:

Maximálisan elérhető pontszám: 100 pont. A sikeres teljesítéshez legalább 51%-os (51 pont) teljesítés szükséges.

50 pontig: elégtelen (1)

62 pontig: elégséges (2)

75 pontig: közepes (3)

88 pontig: jó (4)

100 pontig: jeles (5)

Hallgatói teljesítményértékelő lap

Ellenőrizze, hogy elvégezte-e a tematikus egység valamennyi feladatát! Minden kérdésnél tegyen egy X-et a leginkább megfelelő rovatba, tehát értékelje saját maga a feladat végrehajtását. Ha a felsoroltak közül valamelyik feladat teljesítése nem történt meg vagy lehetetlen volt a teljesítése, tegyen X-et a "Nem" oszlopba.

	Nem	Igen
1. Ismeri az egydimenziós tömbök létrehozásának szintaktikáját.		
2. Ismeri a különböző adattípussal rendelkező tömböket.		
3. Ismeri a bejárásukhoz szükséges szintaktikát.		
4. Ismeri az indexelési szabályokat.		
5. Ismeri a kapcsolatot a ciklusok és a tömbök bejárása között.		

2.7 Kétdimenziós tömbök

2.7.1 Tanulási feladatok

Tartalom

Kétdimenziós tömbök

- Az egydimenziós tömbök kapcsolata a kétdimenziós tömbökkel
- Fűrészfogas tömbök

Kétdimenziós tömbök bejárása

- Kétdimenziós tömbök sor és oszlopindexe
- Sor és oszlop szerinti bejárás

A tematikus egység tanulási eredményei:

A hallgató képes legyen:

- kétdimenziós tömböt létrehozni,
- sor és oszlop szerint bejárni,
- feltölteni tetszőleges adatokkal
- módosítani a tömbben található adatokat
- egy adott tömbelem szomszédos elemeinek a bejárására.

Szükséges eszközök, anyagok:

- A hallgatók felkészüléséhez felhasználható szakirodalom (jegyzet, tankönyv, egyéb források és segédanyagok:

Kötelező:

Ajánlott:

- Coospace-n található órai anyag
- Angster Erzsébet: Objektumorientált tervezés és programozás könyv
- Egyebek:



Tanóra (Kontaktóra) (1 kontaktóra = 45 perc)			Egyéni hallgatói munkaóra óra (1 egyéni hallgatói munkatóra = 60 perc)		
Ren- delke- zésre álló idő- tartam	Tanulási tevékenység	Különleges instrukciók	Ren- delke- zésre álló idő- tartam	Tanulási tevékenység	Különleges instrukciók
2 óra	Ismerje meg a kétdimenziós tömbök különböző létrehozási lehetőségeit! Ismerje meg a sor, illetve oszlop szerinti bejárásokhoz tartozó módszert! Ismerje meg két egymásba ágyazott ciklus viselkedését!	A sor és oszlop szerinti bejárásnál figyeljen a ciklusok megállási feltételére!	3 óra	Készítsen el és töltsön fel adattal egy szorzótáblához tartozó kétdimenziós tömböt! Használja a feltöltött tömböt szorzatok eredményének képzésére!	Ellenőrizze vissza a tömbben található adat helyességét!
1 óra	Készítsen egy példaprogramot amiben szerepel sor és oszlop szerinti bejárás és figyelje meg a különbségeket.				

2.7.2 Önellenőrző feladatok

Magyarázza el az egy és kétdimenziós tömbök közötti különbséget!

Magyarázza el a sor és oszlopindex fogalmát!

Magyarázza el a sor és oszlop szerinti bejárásokat!

Magyarázza el két egymásba ágyazott ciklus viselkedését és kapcsolatát a kétdimenziós tömbökkel!

2.7.3 Megoldókulcs az önellenőrző feladatokhoz

Kifejtette a sor és oszlop elnevezéseket a kétdimenziós tömb kapcsán (20 pont)

Kifejtette, hogy 0-tól kezdődően indexeljük a sor és oszlop elemeit is (40 pont)

Kifejtette két egymásba ágyazott ciklus esetén a ciklusváltozók fontosságát és rámutatott a sor és oszlopindexeként történő használatukra (40 pont)

Az önellenőrzés értékelése:

Maximálisan elérhető pontszám: 100 pont. A sikeres teljesítéshez legalább 51%-os (51 pont) teljesítés szükséges.

50 pontig: elégtelen (1)

62 pontig: elégséges (2)

75 pontig: közepes (3)

88 pontig: jó (4)

100 pontig: jeles (5)

Hallgatói teljesítményértékelő lap

Ellenőrizze, hogy elvégezte-e a tematikus egység valamennyi feladatát! Minden kérdésnél tegyen egy X-et a leginkább megfelelő rovatba, tehát értékelje saját maga a feladat végrehajtását. Ha a felsoroltak közül valamelyik feladat teljesítése nem történt meg vagy lehetetlen volt a teljesítése, tegyen X-et a "Nem" oszlopba.

		Nem	Igen
1.	Ismeri a sor és oszlopindexek fogalmát.		
2.	Ismeri kétdimenziós tömb bejárásának módszereit.		
3.	Ismeri a kétdimenziós tömb létrehozásának szintaktikáját.		
4.	Ismeri a sorok és oszlopok indexelési szabályait.		

2.8 Dinamikus programozás

2.8.1 Tanulási feladatok

Tartalom

Dinamikus programozás

- Egy és többdimenziós táblázatkitöltés
- Dinamikus programozás megoldási módszere
- Bottom-up stratégia
- Sorozatok generálása dinamikus programozással

A tematikus egység tanulási eredményei:

A hallgató képes legyen:

- egy probléma dinamikus programozással történő megoldására
- egy feladat részproblémákra bontására
- a részeredmények tömbökben történő memorizálására

Szükséges eszközök, anyagok:

- A hallgatók felkészüléséhez felhasználható szakirodalom (jegyzet, tankönyv, egyéb források és segédanyagok:

Kötelező:

Ajánlott:

- Coospace-n található órai anyag
- Angster Erzsébet: Objektumorientált tervezés és programozás könyv
- Egyebek:



Tanóra (Kontaktóra) (1 kontaktóra = 45 perc)			Egyéni hallgatói munkaóra óra (1 egyéni hallgatói munkatóra = 60 perc)		
Ren- delke- zésre álló idő- tartam	Tanulási tevékenység	Különleges instrukciók	Ren- delke- zésre álló idő- tartam	Tanulási tevékenység	Különleges instrukciók
3 óra	Ismerje meg a dinamikus programozás alapötletét! Ismerje meg az egy, illetve kétdimenziós tömbökkel való kapcsolatot! Ismerje meg a részproblémákra bontás alaplépéseit! Ismerje meg a tipikus problémákat melyek megoldhatók dinamikus programozással! Ismerje meg a bottom up stratégiát!	A dinamikus programozás során az aktuális elemet mindig meglévő már kiszámolt elemekből származtatjuk!	3 óra	Készítsen egy példaprogramot melyben a dinamikus programozást használja egy feladat megoldására, majd ellenőrizze a táblázatban található adatokat. Hozzon létre a módszert használó sorozatokat egy tömbben!	Figyeljen a helyes tömbbejárásra és indexelésre!

2.8.2 Önellenőrző feladatok

Magyarázza el a dinamikus programozás alapötletét!

Magyarázza el a bottom up stratégia lényegét!

Magyarázza el a megoldási módszer tömbökkel való kapcsolatát!

Mutasson példát problémákra melyek megoldhatók dinamikus programozással!

Magyarázza el egy probléma részproblémákra való bontásának előnyeit!

2.8.3 Megoldókulcs az önellenőrző feladatokhoz

Kifejtette a következő kulcsszavakat: táblázatkitöltés, részproblémára bontás, bottom up (20 pont)

Kifejtette, hogy lentről felfelé építjük a megoldást a bottom up stratégia szerint (20 pont)

Kifejtette, hogy kisebb problémákból felépíthető a végső megoldás (20 pont)

Kifejtette a következő fogalmakat: Fibonacci sorozat, pénzváltás probléma, hátizsák feladat (40 pont)

Az önellenőrzés értékelése:

Maximálisan elérhető pontszám: 100 pont. A sikeres teljesítéshez legalább 51%-os (51 pont) teljesítés szükséges.

50 pontig: elégtelen (1)

62 pontig: elégséges (2)

75 pontig: közepes (3)

88 pontig: jó (4)

100 pontig: jeles (5)

Hallgatói teljesítményértékelő lap

Ellenőrizze, hogy elvégezte-e a tematikus egység valamennyi feladatát! Minden kérdésnél tegyen egy X-et a leginkább megfelelő rovatba, tehát értékelje saját maga a feladat végrehajtását. Ha a felsoroltak közül valamelyik feladat teljesítése nem történt meg vagy lehetetlen volt a teljesítése, tegyen X-et a "Nem" oszlopba.

		Nem	Igen
1.	Ismeri a dinamikus programozás módszerének lényegét.		
2.	Ismeri a bottom up technikát.		
3.	Ismeri a táblázatkitöltés szerepét a megoldásban.		
4.	Ismeri egy probléma részproblémákra történő bontásának menetét.		

2.9 Keresés, kiválasztás, rendezés

2.9.1 Tanulási feladatok

Tartalom

Keresés

- Elemek keresése tömbökben
- Leszámlálás
- Kiválogatás

Kiválasztás

- Minimum kiválasztás
- Maximum kiválasztás

Rendezés

- Minimumkiválasztásos rendezés
- Beszűrőrendezés
- Buborékrendezés

A tematikus egység tanulási eredményei:

A hallgató képes legyen:

- elemek után keresni és feltételek leválogatni vagy megszámolni egy adott tömb elemeit
- minimum és maximum elemek meghatározására egy tömbben
- rendezni egy tömböt különböző módszerekkel

Szükséges eszközök, anyagok:

- A hallgatók felkészüléséhez felhasználható szakirodalom (jegyzet, tankönyv, egyéb források és segédanyagok:

Kötelező:

Ajánlott:

- Coospace-n található órai anyag
- Angster Erzsébet: Objektumorientált tervezés és programozás könyv
- Egyebek:



Tanóra (Kontaktóra) (1 kontaktóra = 45 perc)			Egyéni hallgatói munkaóra óra (1 egyéni hallgatói munkatúra = 60 perc)		
Ren- delke- zésre álló idő- tartam	Tanulási tevékenység	Különleges instrukciók	Ren- delke- zésre álló idő- tartam	Tanulási tevékenység	Különleges instrukciók
3 óra	Ismerje meg a tömbök elemeinek keresését, illetve valamilyen feltétel szerinti leválogatását és megszámlálását! Ismerje meg a minimum és maximum elemek keresésének módszerét! Ismerje meg a különböző rendezések közötti különbségeket és hatékonyságukat!	A rendezésnél figyelje meg az elemek helyét minden egyes lépés után!	3 óra	Készítsen egy otthoni projektet mely bemutatja az egyes tanult módszereket! Gyakorolja a különböző módszerek szintaktikáját és törekedjen a megértésükre!	A kiválasztások és keresések esetén érdemes a módszer minden lépését tanulmányozni a működésének megértéséhez!

2.9.2 Önellenőrző feladatok

Magyarázza el a leszámolás, illetve kiválogatás fogalmakat!
Magyarázza el az elemek keresésének folyamatát a tömbökben!
Magyarázza el a minimum, illetve maximumkeresések menetét!
Magyarázza el a rendezések célját, illetve módszereit!

2.9.3 Megoldókulcs az önellenőrző feladatokhoz

Kifejtette a következő fogalmakat: adott feltételnek megfelelő elemek megszámlálása, és kigyűjtése (20 pont)

Kifejtette egy adott elem keresésének és eredményként történő megtalálásának a menetét (20 pont)

Kifejtette a minimum és maximum keresések során használt feltételeket (20 pont)

Kifejtette a min és max változók használatát a minimum és maximum keresések során (20 pont)

Kifejtette a következő fogalmakat magyarázattal együtt: beszűrő rendezés, buborék rendezés, minimumkiválasztásos rendezés (20 pont)

Az önellenőrzés értékelése:

Maximálisan elérhető pontszám: 100 pont. A sikeres teljesítéshez legalább 51%-os (51 pont) teljesítés szükséges.

50 pontig: elégtelen (1)

62 pontig: elégséges (2)

75 pontig: közepes (3)

88 pontig: jó (4)

100 pontig: jeles (5)

Hallgatói teljesítményértékelő lap

Ellenőrizze, hogy elvégezte-e a tematikus egység valamennyi feladatát! Minden kérdésnél tegyen egy X-et a leginkább megfelelő rovatba, tehát értékelje saját maga a feladat végrehajtását. Ha a felsoroltak közül valamelyik feladat teljesítése nem történt meg vagy lehetetlen volt a teljesítése, tegyen X-et a "Nem" oszlopba.

	Nem	Igen
1. Ismeri a keresések, leválogatások és megszámlálások menetét.		
2. Ismeri a minimum és maximum keresések folyamatát.		
3. Ismeri az egyes rendezések menetét és hatékonyságukat.		
4. Ismeri a tanult módszerekhez tartozó kódrészleteket.		

2.10 Metódusok

2.10.1 Tanulási feladatok

Tartalom

Metódusok

- Metódusok és függvények
- Paraméterek
- Visszatérési érték
- Függvényhívás

A tematikus egység tanulási eredményei:

A hallgató képes legyen:

- metódusok és függvények létrehozására
- paraméterek definiálására és használatára
- visszatérési értékkel rendelkező metódusok létrehozására
- megírt függvények meghívására

Szükséges eszközök, anyagok:

- A hallgatók felkészüléséhez felhasználható szakirodalom (jegyzet, tankönyv, egyéb források és segédanyagok):

Kötelező:

Ajánlott:

- Coospace-n található órai anyag
- Angster Erzsébet: Objektumorientált tervezés és programozás könyv
- Egyebek:



Tanóra (Kontaktóra) (1 kontaktóra = 45 perc)			Egyéni hallgatói munkaóra óra (1 egyéni hallgatói munkatóra = 60 perc)		
Ren- delke- zésre álló idő- tartam	Tanulási tevékenység	Különleges instrukciók	Ren- delke- zésre álló idő- tartam	Tanulási tevékenység	Különleges instrukciók
2 óra	Ismerje meg az egyes kódrészletek átírását metódus vagy függvényformába! Ismerje meg a paraméterek működését és a láthatóságukat a metódusokban!	A kódismétlések minimalizálására érdemes használni metódusokat!	3 óra	Készítse el az eddig tanult programokat úgy, hogy használja a metódusokat a kódismétlések eltüntetésére! Készítsen olyan kódot, amely tetszőleges paraméterrel működik!	A kódismétlések minimalizálására érdemes használni metódusokat!
1 óra	Ismerje meg a visszatérési értékek működését!				

2.10.2 Önellenőrző feladatok

Magyarázza el a metódusok működését!

Magyarázza el a paraméterek értelmét és a viselkedését a metódushíváskor!

Magyarázza el a visszatérési érték szerepét és használatát a metódushíváskor!

Magyarázza el a metódusok szerepét a programozásban!

2.10.3 Megoldókulcs az önellenőrző feladatokhoz

Kifejtette a metódusok kapcsán a következő fogalmakat: paraméter, visszatérési érték, kódismétlés (20 pont)

Kifejtette, hogy a paraméterek használata általánosíthatóvá tesznek bizonyos kódrészleteket. (20 pont)

Ismertette a paraméterek szintaktikáját és azt, hogy lokális változóként viselkednek a metódusban melyek a metódushíváskor kapnak értéket. (20 pont)

Kifejtette, hogy a visszatérési érték szerepe az eredmények visszaadása a híváskor. (20 pont)

Kifejtette hogy a metódusok használatával csökkenthető a kódismétlés és a kódminőség. (20 pont)

Az önellenőrzés értékelése:

Maximálisan elérhető pontszám: 100 pont. A sikeres teljesítéshez legalább 51%-os (51 pont) teljesítés szükséges.

50 pontig: elégtelen (1)

62 pontig: elégséges (2)

75 pontig: közepes (3)

88 pontig: jó (4)

100 pontig: jeles (5)

Hallgatói teljesítményértékelő lap

Ellenőrizze, hogy elvégezte-e a tematikus egység valamennyi feladatát! Minden kérdésnél tegyen egy X-et a leginkább megfelelő rovatba, tehát értékelje saját maga a feladat végrehajtását. Ha a felsoroltak közül valamelyik feladat teljesítése nem történt meg vagy lehetetlen volt a teljesítése, tegyen X-et a "Nem" oszlopba.

	Nem	Igen
1. Ismeri a metódusok szintaktikáját és használatát.		
2. Ismeri a paraméterek szerepét a metódushíváskor.		
3. Ismeri a visszatérési érték szerepét a metódushíváskor.		
4. Ismeri a kódismétlés fogalmát és a metódusokkal való kapcsolatát.		



2.11 Metódusok lokális változói

2.11.1 Tanulási feladatok

Tartalom

Lokális és globális változók

- Láthatóság fogalma
- Globális változók
- Lokális változók
- Lokális változók kapcsolata a metódusokkal

A tematikus egység tanulási eredményei:

A hallgató képes legyen:

- globális és lokális változókat használni
- a globális és lokális változók közötti különbségek ismertetésére
- meghatározni egy változó láthatóságát egy programban

Szükséges eszközök, anyagok:

- A hallgatók felkészüléséhez felhasználható szakirodalom (jegyzet, tankönyv, egyéb források és segédanyagok:

Kötelező:

Ajánlott:

- Coospace-n található órai anyag
- Angster Erzsébet: Objektumorientált tervezés és programozás könyv
- Egyebek:





Tanóra (Kontaktóra) (1 kontaktóra = 45 perc)			Egyéni hallgatói munkaóra óra (1 egyéni hallgatói munkatóra = 60 perc)		
Ren- delke- zésre álló idő- tartam	Tanulási tevékenység	Különleges instrukciók	Ren- delke- zésre álló idő- tartam	Tanulási tevékenység	Különleges instrukciók
3 óra	Ismerje meg a globális változók fogalmát és láthatóságukat a programon belül! Ismerje meg a lokális változók fogalmát és láthatóságukat a metódusokon és blokkokon keresztül! Ismerje meg a lokális változók kapcsolatát a metódusokkal! Ismerje meg a láthatóság fogalmát és szerepét a programok írása során!	Figyeljen rá, hogy lokális változót nem lehet elérni onnan, ahol nem látható!	3 óra	Készítsen lokális és globális változókat a programban, és próbálja ki mely programrészekben élnek a változók és melyekben nem! Készítsen metódust melyben vannak lokális változók és próbálja meg elérni őket a metóduson kívül!	Figyeljen rá, hogy lokális változót nem lehet elérni onnan, ahol nem látható!

2.11.2 Önellenőrző feladatok

Magyarázza el a láthatóság fogalmát!

Magyarázza el a globális változók láthatóságát és szerepét a programozásban!

Magyarázza el a lokális változók láthatóságát és kapcsolatukat a metódusokkal!

Magyarázza el a metódusok paramétereinek a láthatóságát!

2.11.3 Megoldókulcs az önellenőrző feladatokhoz

Kifejtette, hogy a láthatóság a változók elérhetőségét jelenti a programon belül. (30 pont)

Kifejtette, hogy a globális változók a létrehozásuk után mindenhol láthatók a programon belül. (20 pont)

Kifejtette, hogy a lokális változók csak a saját blokkjukban láthatók. (20 pont)

Kifejtette, hogy a metódusok paraméterei a metóduson belül láthatók így lokális változóként működnek. (30 pont)

Az önellenőrzés értékelése:

Maximálisan elérhető pontszám: 100 pont. A sikeres teljesítéshez legalább 51%-os (51 pont) teljesítés szükséges.

50 pontig: elégtelen (1)

62 pontig: elégséges (2)

75 pontig: közepes (3)

88 pontig: jó (4)

100 pontig: jeles (5)

Hallgatói teljesítményértékelő lap

Ellenőrizze, hogy elvégezte-e a tematikus egység valamennyi feladatát! Minden kérdésnél tegyen egy X-et a leginkább megfelelő rovatba, tehát értékelje saját maga a feladat végrehajtását. Ha a felsoroltak közül valamelyik feladat teljesítése nem történt meg vagy lehetetlen volt a teljesítése, tegyen X-et a "Nem" oszlopba.

	Nem	Igen
1. Ismeri a globális változók szerepét és használatukat.		
2. Ismeri a lokális változók használatát és kapcsolatukat a metódusokkal.		
3. Ismeri a globális és lokális változók közti különbséget.		
4. Ismeri a láthatóság fogalmát.		
5. Ismeri egy tetszőleges változó láthatóságát a programon belül.		

2.12 Rekurzió és rekurzív problémák

2.12.1 Tanulási feladatok

Tartalom

Metódusok

- Rekurzió fogalma
- Függvények rekurzív hívása
- Végtelen rekurziók
- Rekurzív megállási feltételek

Rekurzív problémák

- Rekurzívan megoldható problémák
- Rekurzív problémák megoldásának módszere

A tematikus egység tanulási eredményei:

A hallgató képes legyen:

- a rekurzió alkalmazására és rekurzív algoritmusok létrehozására
- rekurzív feladatok megoldására
- saját rekurzív megoldási módszer fejlesztésére
- dinamikus programozás és rekurzió kapcsolatának felismerésére

Szükséges eszközök, anyagok:

- A hallgatók felkészüléséhez felhasználható szakirodalom (jegyzet, tankönyv, egyéb források és segédanyagok:

Kötelező:

Ajánlott:

- Coospace-n található órai anyag
- Angster Erzsébet: Objektumorientált tervezés és programozás könyv
- Egyebek:



Tanóra (Kontaktóra) (1 kontaktóra = 45 perc)			Egyéni hallgatói munkaóra óra (1 egyéni hallgatói munkatóra = 60 perc)		
Ren- delke- zésre álló idő- tartam	Tanulási tevékenység	Különleges instrukciók	Ren- delke- zésre álló idő- tartam	Tanulási tevékenység	Különleges instrukciók
3 óra	Ismerje meg a rekurzió fogalmát és a rekurzív hívások sorrendjét! Ismerje meg a rekurzívan megoldható problémákat és megoldási módszerüket! Ismerje meg a végtelen és megállási feltétellel rendelkező rekurziók természetét! Ismerje meg a rekurzív megoldás létrehozásának a módszerét egy problémára!	Egy megoldás rekurzív építésénél érdemes felírni a hívások során visszaadott értékeket egyesével!	3 óra	Hozzon létre végtelen és nem végtelen rekurziókat! Programozzon le egy rekurzív megoldási módszert egy tetszőleges problémára és értse meg a rekurzió sorrendjét és részeredményeit! Ismerje meg a veremtúlcsoordulást végtelen rekurziók esetén!	Egy megoldás rekurzív építésénél érdemes felírni a hívások során visszaadott értékeket egyesével!

2.12.2 Önellenőrző feladatok

Magyarázza el a rekurzió fogalmát és szerepét a programozásban!
Magyarázza el a végtelen rekurzió és a megállási feltétel fogalmát!
Magyarázza el a rekurzív hívások sorrendjét!
Magyarázza el a veremtúlsordulás jelenségét!
Ismertessen legalább egy rekurzív problémát megoldással együtt!

2.12.3 Megoldókulcs az önellenőrző feladatokhoz

Kifejtette, hogy a rekurzió során metódusok önmagukat hívják meg és ismertette, hogy megállási feltétel nélkül végtelen rekurziót kapunk (20 pont)
Kifejtette, hogy a megállási feltétel során a metódus nem hívja tovább önmagát (20 pont)
Kifejtette, hogy végtelen rekurzió esetén veremtúlsordulás fordulhat elő (20 pont)
Kifejtette a faktoriális számítás vagy más rekurzív problémát megoldással együtt (40 pont)

Az önellenőrzés értékelése:

Maximálisan elérhető pontszám: 100 pont. A sikeres teljesítéshez legalább 51%-os (51 pont) teljesítés szükséges.

50 pontig: elégtelen (1)

62 pontig: elégséges (2)

75 pontig: közepes (3)

88 pontig: jó (4)

100 pontig: jeles (5)

Hallgatói teljesítményértékelő lap

Ellenőrizze, hogy elvégezte-e a tematikus egység valamennyi feladatát! Minden kérdésnél tegyen egy X-et a leginkább megfelelő rovatba, tehát értékelje saját maga a feladat végrehajtását. Ha a felsoroltak közül valamelyik feladat teljesítése nem történt meg vagy lehetetlen volt a teljesítése, tegyen X-et a "Nem" oszlopba.

		Nem	Igen
1.	Ismeri a rekurzió működését és képes használni.		
2.	Ismeri a végtelen rekurzió fogalmát és tudja mi az a megállási feltétel.		
3.	Ismeri a veremtúlsordulás jelenségét.		
4.	Ismeri a rekurzív problémákat.		

2.13 Objektumorientáltság alapjai

2.13.1 Tanulási feladatok

Tartalom

Objektumorientáltság

- Osztályok és objektumok
- Konstruktorok
- Osztályok attribútumai
- Osztályok metódusai
- Példánymetódus
- Statikus attribútumok és metódusok

A tematikus egység tanulási eredményei:

A hallgató képes legyen:

- osztályok definiálására
- osztályokból objektumok példányosítására
- osztályattribútumok

Szükséges eszközök, anyagok:

- A hallgatók felkészüléséhez felhasználható szakirodalom (jegyzet, tankönyv, egyéb források és segédanyagok):

Kötelező:

Ajánlott:

- Coospace-n található órai anyag
- Angster Erzsébet: Objektumorientált tervezés és programozás könyv
- Egyebek:



Tanóra (Kontaktóra) (1 kontaktóra = 45 perc)			Egyéni hallgatói munkaóra óra (1 egyéni hallgatói munkatóra = 60 perc)		
Ren- delke- zésre álló idő- tartam	Tanulási tevékenység	Különleges instrukciók	Ren- delke- zésre álló idő- tartam	Tanulási tevékenység	Különleges instrukciók
2 óra	Ismerje meg az objektumorientált paradigma lényegét! Ismerje meg az osztályok szerepét a programozásban! Ismerje meg az attribútumok szerepét és használatukat az osztályon belül! Ismerje meg a konstruktor működését és szerepét! Ismerje meg az osztálymetódusokat és szerepüket az osztályon belül!	Az osztály tervrajzként tehát egyfajta leírásként működik, míg az objektum egy konkrét példánya az osztálynak!	2 óra	Készítsen osztályt egy valóságból származó példán keresztül, mely rendelkezik attribútumokkal, konstruktorral, valamint metódusokkal! Olvasson utána a vonatkozó résznek az Angster Erzsébet: Objektumorientált tervezés és programozás könyvben! Próbálja ki a konstruktorok működését különböző paraméterlistával!	Az objektum példányosítása a konstruktoron keresztül történik, azonban konstruktorból lehet több is különböző paraméterlistával!
1 óra	Ismerje meg az objektumok létrehozását osztályokból! Ismerje meg a statikus osztálymetódusokat és attribútumokat, valamint a viselkedésüket!	Az osztály tervrajzként tehát egyfajta leírásként működik, míg az objektum egy konkrét példánya az osztálynak!	1 óra	Példányosítson objektumokat a létrehozott osztályból és figyelje meg a különböző tulajdonságokkal felruházott példányokat!	Amennyiben üres paraméterlistával példányosít egy objektumot, úgy a default konstruktorban definiált értékekkel jön létre a példány!

2.13.2 Önellenőrző feladatok

Ismertesse az objektumorientált paradigma alapelveit!

Ismertesse a különbséget osztály és objektum között!

Készítsen egy osztályt az órán tanult szemléletmód alapján!

Készítse el a main függvényt az osztályhoz, és használja az előzőekben létrehozott osztályt, példányosítson belőle objektumokat!

Sorolja fel az osztály főbb alkotóelemeit!

Hozzon létre statikus metódusokat és attribútumokat az osztályban!

2.13.3 Megoldókulcs az önellenőrző feladatokhoz

Kifejtette, a következő kulcsszavakat: absztrakció, egységbezárás, öröklődés, polimorfizmus, kódújr felhasználás, adatok elrejtése (20 pont)

Kifejtette, hogy az osztályból származnak az objektumok, melyek konkrét példányokat jelentenek (20 pont)

Szintaktikailag helyes programot sikerült írnia melyben található egy osztály, amelyet a main függvényben használ (40 pont)

Statikus metódusokkal és attribútumokkal sikeresen bővítette a programot (20 pont)

Az önellenőrzés értékelése:

Maximálisan elérhető pontszám: 100 pont. A sikeres teljesítéshez legalább 51%-os (51 pont) teljesítés szükséges.

50 pontig: elégtelen (1)

62 pontig: elégséges (2)

75 pontig: közepes (3)

88 pontig: jó (4)

100 pontig: jeles (5)

Hallgatói teljesítményértékelő lap

Ellenőrizze, hogy elvégezte-e a tematikus egység valamennyi feladatát! Minden kérdésnél tegyen egy X-et a leginkább megfelelő rovatba, tehát értékelje saját maga a feladat végrehajtását. Ha a felsoroltak közül valamelyik feladat teljesítése nem történt meg vagy lehetetlen volt a teljesítése, tegyen X-et a "Nem" oszlopba.

	Nem	Igen
1. Ismeri az objektumorientált paradigma alapelveit.		
2. Tisztában van a szintaktikailag helyes osztálydefiníció szabályaival.		
3. Ismeri és el tudja különíteni az osztály és objektum fogalmakat.		
4. Képes meglévő osztályokból példányokat létrehozni és használni őket.		
5. Fel tudja sorolni az osztály részeit és szerepüket.		
6. Ismeri a statikus metódusokat és attribútumokat.		