

Programozás I. Nagy zh

SZTE Szoftverfejlesztés Tanszék

2023. tavasz

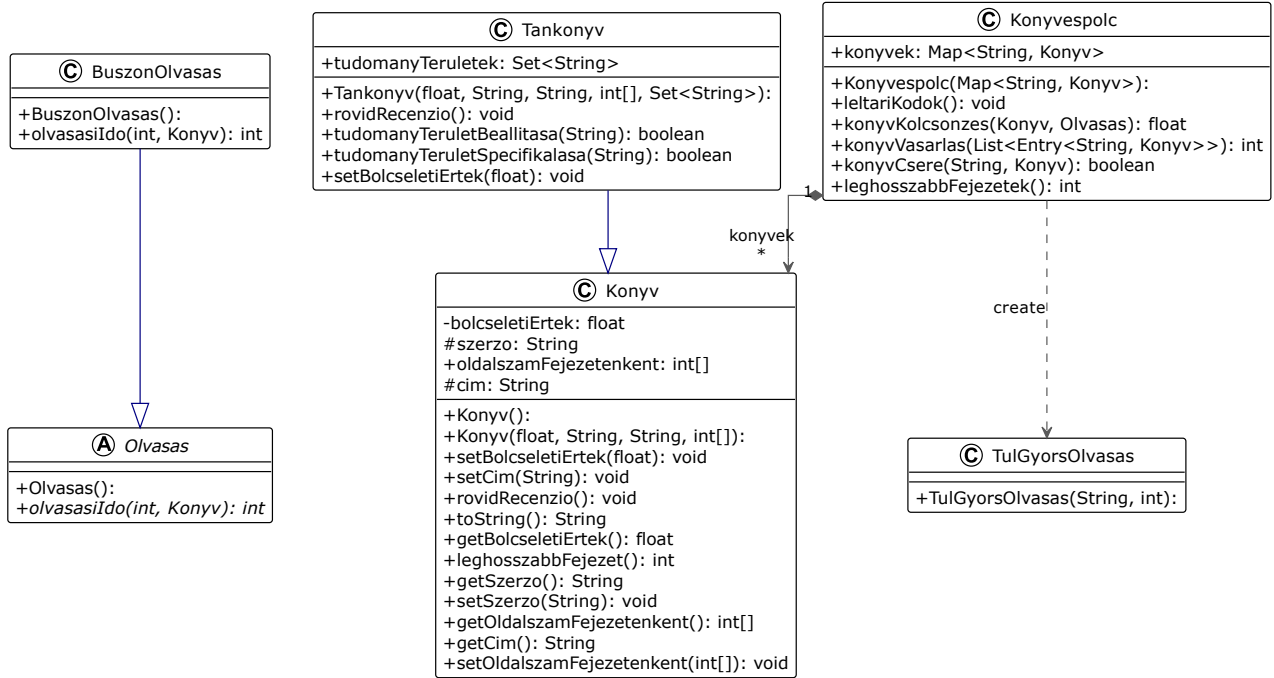
Általános követelmények, tudnivalók

- A feladat elkészítésére 90 perc áll a rendelkezésre. Ez szigorú határidő, a Bíró előre megadott időben zár.
- A feladatokat számítógép előtt kell megoldani, tetszőleges fejlesztői környezetben, tetszőleges operációs rendszer segítségével.
- Az elkészült programot **20** alkalommal lehet benyújtani, a megadott határidőig.
- Csak a leírásban szereplő osztályokat, metódusokat és adattagokat kell megvalósítani, egyéb dolgokért nem jár plusz pont.
 - Aki Windowst használ, annak a gép elindítása után érdemes egyből a fejlesztőkörnyezetet elindítani, és létrehozni egy új projektet, és csak utána a böngészőt, mivel az elején egy néhány percig indexel, addig pont el lehet olvasni a feladatot.
- Bármely segédanyag használata **tilos** (a fejlesztőkörnyezetek nyújtotta segítségen kívül), aki mégis ilyet tesz, vagy próbálkozik vele, annak a dolgozata nem értékelhető és a kurzus nem teljesített. Ha valakinek a padtársa segít, akkor mérlegelés nélkül mindkettő hallgató dolgozata sikertelen, a kurzus nem teljesített.
- A feladat megoldása során minden megadott előírást pontosan követni kell! Tehát, ha a feladat leírása szerint egy adattag neve a "elsoFoku", akkor az alábbi elnevezések nem megfelelőek: "elsőFokú", "elsofoku", "elso_foku", "elsőFoq". Ugyanez igaz a metódusok, osztályok elnevezésére is!
- A metódusok esetében a visszatérési típus, a név, módosítók és a paraméterek típusai (és azok sorrendje) kerülnek ellenőrzésre, azonban a paraméterek nevei tetszőlegesek lehetnek.
- Az órán tanult konvenciókat követni kell (getter/setter elnevezés, toString, indentálás, stb). Abban az esetben is, ha ezt a feladat külön nem emeli ki, az ellenőrzés során erre is építünk.
- A nem forduló kódok nem kerülnek kiértékelésre, ezt utólagosan a gyakorlatvezető sem bírálhatja felül. (Hiszen mindenki rendelkezésére áll a saját környezete, ahol fordítani, futtatni tudja a forráskódot, így feltöltés előtt ezt mindenképpen érdemes megnézni!)
- Az adattagok és konstruktorok hiányában garantáltan 0 pontos lesz a kiértékelés, mert ezek minden teszt alapját képezik.
- Ha végtelen ciklus van a programban, akkor ezt a Bíró ki fogja dobni 3 másodperc után (ha többször is meghívásra kerül ilyen metódus, akkor ez többszöri 3 másodperc, összesen akár 2 perc is lehet). Ilyenkor NE kattints még egyszer a *Feltöltés* gombra, mert akkor

kifagyhat a Bíró, csak a böngésző újraindításával lehet megoldani a problémát (emellett elveszik 1 feltöltési lehetőség is).

- Kérdés/probléma esetén a gyakorlatvezetők tudnak segítséget nyújtani.
- **A feladat megoldása során a default csomagba dolgozz, majd a kész forrásfájlokat tömörítve, zip formátumban töltsd fel, azonban a zip fájlt tetszőlegesen elnevezheted!**
 - Zip készítése: Windowson és Linuxon is lehet a GUI-ban jobb klikkes módszerrel tömörített állományt létrehozni (Windowsban pl. a 7-Zip nevű ingyenes program használatával).
 - Linux terminálon belül például a "zip feladat.zip *.java" paranccsal is elkészíthető a megfelelő állomány.
- A feladatokban az alábbi dolgok az alapértelmezettek (**kivéve**, ha a feladat szövege mást mond)
 - az osztályok láthatósága publikus
 - az egész érték 32 bites
 - a lebegőpontos számok dupla pontosságúak
 - az olyan metódusok void visszatéréssel rendelkeznek, amelyeknél nincs specifikálva visszatérési típus.
 - a metódusok mindenki számára láthatóak
 - az adattagok csak az adott osztályban legyenek elérhetőek
- A *riport.txt* és a fordítási log fájlok megtekinthetőek az alábbi módon:
 1. Az *Eredmények megtekintése* felületen a vizsgálandó próba új lapon való megnyitása
 2. A kapott url formátuma:
`https://biro.inf.u-szeged.hu/Hallg/IB204L-1/1/hXXXXXX/4/riport.txt`
 3. Az url-ből visszatörölve a 4-esig (*riport.txt* törlése) megkaphatók a 4-es próbálkozás adatai.
- Szövegek összehasonlításánál az egyezés a pontos egyezést jelenti, azaz ha kis-nagy betűben térnek el, akkor már nem tekinthetők egyenlőnek (pl. a "piros" != "Piros")
- A leírásokban bemutat példákban a stringek köré rakott idézőjelek nem részei az elvárt kimenetnek, azok csak a string határait jelölik. Például ha az szerepel, hogy "3 alma", akkor az elvárt kimenet idézőjelek nélkül az 3 alma, de a szóköz szükséges!
- Az elvárt kimeneteknek karakterről karakterre olyan formátumúnak kell lennie, ami a feladatban le van írva (szóközöket és sortöréseket is beleértve).

1. ábra. Osztálydiagram



1. feladat Konyv (12 pont)

A könyv a tudás forrása, a legtöbb kultúrában tisztelettel övezik, és a bölcsesség szimbólumának használják. Jelen zshban viszont ez most mindössze egy osztály, mely ha az létezik, helyes fejléccel és módosítókkal, akkor pontot ér. (1 pont)

Konstruktorok

Default konstruktor (1 pont): a könyv, nem mint papír, de mint nyomtatott szó, bölcsességet hordoz, melyet nehéz számszerűsíteni, de digitális korunkban sajnos muszáj. A könyv bölcséleti értékét nagy általánosságban határozzuk meg 10-ben, ha pedig nem tudjuk a címét, akkor a címet állítsuk „ismeretlen kotet”-re. A szerzővel járjunk el hasonló módon, bár itt a történelem más példákat is említ, hogy csak a magyar gyökerekkel rendelkező Anonimusra utaljak, de jelen helyzetben elégedjünk meg az "ismeretlen szerzo" kifejezéssel. Egy ismeretlen könyv fejezeteinek a számát meghatározni elég nehéz, itt az vezessen minket, hogy 80 fejezet bőven elég mindenre.

Paraméteres konstruktor (2 pont): Ha a könyv adatai a rendelkezésünkre állnak, akkor persze sokkal könnyebb dolgunk van. Egyszerűen csak vegyük sorban az érkező adatokat, és illesszük azokat a megfelelő adattagokhoz. Talán érdemes megjegyezni, hogy ez egy különleges alkalommal a cím megelőzi a szerzőt. A számsort, mely a fejezetek oldalszámait tartalmazza, ne legyünk restek számról számra bemásolni a megfelelő helyre! Mint már utaltam rá, a könyv által hordozott bölcsesség nehezen számszerűsíthető. Ám bizonyos felvetésekkel élhetünk! Egyezzünk meg abban, hogy bölcséleti érték nem lehet kevesebb, mint a könyv fejezetei számának kétszerese, hiszen egy hosszú könyv biztosan tartalmaz valami okosságot is!

Adattagok

A fentebb taglalt adattagok létezzenek, és legyen megfelelő a láthatóságuk. (1 pont)

Metódusok

Getterek és Setterek (2 pont): a bölcséleti értékre tett feltevésünket, melyet már a paraméteres konstruktorban is alkalmaztunk, továbbra is tartsuk szem előtt, de csak akkor, ha őt magát változtatnánk. Ha a környezetében változnak a többi adattagok, akkor őt magát, külön utasítás híján, hagyjuk békén!

toString (1 pont): ez a kis metódus praktikus összefoglalót ad a könyvről. Rövidke szöveggel tér vissza, mely a következőképpen hangzik: "A(z) <cim> cimu konyv <milyen> olvasmany.". Természetesen a cím-címke helyére a tényleges címet kell beilleszteni, a milyenség pedig ez alkalommal egy szóba sűrítettik össze. "konnyed" vagy "sulyos" lehet, attól függően, hogy a bölcséleti érték hogyan alakul. Ha kisebb mint 20, akkor inkább szórakoztató olvasmányról beszélhetünk, egyéb esetben fajsúlyosabbról.

rovidRecenzio (2 pont): a könyvek általában kritikákat kapnak, ami jó esetben az író épülésére szolgál, de az olvasót is eligazítja. Ez a rövid kis recenzió inkább csak első támpontnak szolgálhat. Ha a kötet bölcséleti értéke elegendően nagy, és itt megint belebotlunk a számszerűsítés problémájába, de határozzuk meg az elegendőt ez alkalommal 100-ban, tehát ilyen helyzetben a metódus kiírja a standard outputra, hogy "A(z) <cim> egy kivalo konyv.". Ellenkező esetben hibacsatornára írja ki, hogy "A(z) <cim> egy konnyed ponyva.", ami elszett ítéletnek hangozhat, de nem feltétlenül negatív! Szórakoztató irodalomra is nagy szüksége van a világnak! Hogy szavam közben ne feledjem, a cím-címke helyére ez alkalommal is a könyv címét kell beilleszteni.

leghosszabbFejezet(**2 pont**): a metódus neve önmagáért beszél. A könyv leghosszabb fejezetének a hosszát kell megadnia. Talán csak az az eset hordoz némi érdekességet, ha nincsenek fejezetek. Ebben a roppant extrém esetben legyen a fiktív leghosszabb fejezet hossza 0.

2. feladat Tankönyv (12 pont)

A Tankönyvet bemutatni talán felesleges is egyetemi hallgatóságnak. Mindenki találkozott már vele. Ez egy különleges könyv, és ebben a feladatban ha osztályként létezik, természetesen helyes fejléccel, akkor pontot ér. (1 pont)

Konstruktorok

Paraméteres konstruktor (1 pont): Amit a könyv tud, tudja a tankönyv is, és ami értékekkel felruházhattuk a könyvet, azokat most is hasonló módon alkalmazhatjuk a tankönyv konstruktorában. Az érkező paraméterek sorában némi érdeklődésre talán csak az utolsó elem tarthat számot. Ez hordozza a tudományTerület adattag leendő értékeit. Ezt az adattagot legvadabb szeszélyeink szerint hozhatjuk létre, persze csak a Java nyelv szigorú szabályait betartva, és mikor elkészültünk vele, itt is másoljuk az adatokat, ne csak hanyagul utalgassunk rá, hogy valahol, a memória hádészi mélységeiben megvannak azok!

Adattagok

Ezen a ponton tehát az adattagoknak létezniük kell, jó láthatósággal, szépen és rendben (1 pont)

Metódusok

rovidRecenzio (2 pont) : amit a könyv kapcsán elmeséltem, az itt is megállja a helyét, mindössze egy apró módosítást kell tennünk. A tankönyv sokkal inkább a bölcsesség forrása, mint a könyv, a szabályai szigorúbban kötik. Az áltudományokat ki kell zárni belőlük, ezek közül pedig a legmélyebb mótely az „ezoterika”. Ha a könyv címe magában hordozza ezt a förtelmet, akkor azonnal cselekednünk kell! A címet ki kell egészíteni a "(pamflet)" kifejezéssel, így óva meg az olvasókat a lesből támadó butasággal szemben.

tudomanyTeruletBeallitasa (3 pont) : a jó könyvtáros első dolga, hogy a tankönyveket a megfelelő helyre illessze a benne foglalt tudományterületek alapján. Ez a kis metódus ebben segít neki azzal, hogy a tankönyv tudományterületeit meghatározza. Ha a paraméterben kapott tudományterület már tudottan ennek a tankönyvnek sajátja, akkor már nem kell semmit tennie, és e tétlenséget szimbolizálandó hamis értékkel tér vissza. Ám ha a kapott tudományterület még nem tartozik a tankönyvhöz, akkor sietve hozzárendeli, majd a jól végzett munka utáni elégedettséggel és egy igaz értékkel tér vissza.

tudomanyTeruletSpecifikalasa (3 pont): a tudomány, mint maga a világ, fejlődik. A változásokat pedig nyomon kell követnünk. Ha a tudomány specializálódik, avagy részeiben változik, akkor ezt a tankönyveinkben is jelölni kell. Ez a kis metódus egy tudományterületet kap, ami ha már jellemzője a tankönyvnek, akkor kiegészítették a következő módon: a "(specifikalt)" kifejezést kell mögé írni, és ezzel a munkánkat igaz módon el is végeztük. Ha a kapott tudományterület nem volt eddig hozzárendelve a tankönyvhöz, akkor talán hamis adatokat kaptunk, vagy valaki valahol valami más hibát vétett. Ekkor nem tehetünk semmit, csak visszaadunk egy hamis értéket, neheztelésünk kifejezése végett.

setBolcseletiErtek (1 pont): a tankönyvnél elvárás, hogy tele legyen okossággal, tehát a bölcséleti értéknek alsó határa nem a fejezetek számának gyarló duplája, de egyenesen négyszerese! Nem csekély vállalás ez! S ha kishitű módon ennél kisebb paramétert kap a függvény, én mondom, kacagjon a világ szemébe, és bátran állítsa a bölcséleti értéket a megfelelő magasságba! Más különlegességről itt nem kell beszélnem, a tankönyv is csak könyv, aminek maradék szabályai reá is vonatkoznak.

3. feladat Olvasas (2 pont)

Az olvasás mint tevékenység szorosan kötődik a könyvekhez. Ha példányt nem is, de osztályt mindenképpen ér ebben a feladatban. Létezzen tehát, helyes fejléccel, büszkén és emelt fővel! (1 pont)

Konstruktorok

E ponton súlyos hiányosságról vagyok kénytelen szólni. Konstruktor, hát, az nincs! De ezzel együtt kell élnünk, és nézzük a jó oldalát! Ennyivel is kevesebb a kódolandó feladat, mi előttünk áll!

Adattagok

Az adattagokat is hiába keressük az UML útvesztőjében. Nincsenek azok sem. De ettől az olvasás még olvasás marad, senki ne csüggedjen!

Metódusok

olvasasiIdo (1 pont) : és egy metódus, amiről még nem lehet beszélni. Csupán ígéret ő, melyet a következő osztályban, itt, alatt, mindjárt megvalósítunk.

4. feladat BuszonOLvasas (3 pont)

Az extrém olvasás még nem sport, de mondom, lehetne! És ha nagyon nem is akarunk belelendülni, különleges helyzetekben azért szoktunk olvasni. Itt most azt az alkalmat emelném ki, amikor a tömegközlekedés rázkódó valóságában kénytelen az ember kibogarászni a betűket. Ez a BuszonOlvasas, mely kétségtelenül létezik, mint az olvasás egy válfaja, és hamarosan leírásra kerül! (1 pont)

Konstruktorok

Újabb fájó hiány! Sehol egy konstruktor. Üres világban élünk!

Adattagok

És adattagok? Csak a szél kavarja a port a puszta felett, hová múltott el a világ dicsősége?

Metódusok

olvasasiIdo (2 pont) : és mint ahogy mondani szokták, a jó kolléga nem csak ígér, de be is tart! Az előző osztályban emlegetett olvasasiIdo megvalósításáról beszélek. Az olvasási idő, hogy fizikusoknak tetsző módon fogalmazzuk meg, egyenesen arányos a kapott könyv fejezeteinek a számával, és szintén egyenesen arányos az első paraméterben érkező olvasótempóval.

5. feladat TulGyorsOlvasas (3 pont)

A világ veszélyekkel teli, és bizony még egy egyszerű, délutáni kis olvasásból is lehet baj! Ezeknek a bajoknak a kiküszöbölésére alkossunk kivétel osztályt! Természetesen helyes fejléccel, és általában, a szakma szabályai szerint. **(1 pont)**

Konstruktorok

Paraméteres konstruktor (2 pont): ez a kivételosztály, mint fajtájának legtöbbje, csak szavakkal figyelmeztet. Ha a kapott értékek alapján úgy találja, a baj nem túl nagy, akkor a kapott üzenettel szalad tovább a felmenője felé. Ha viszont súlyos, fekete felhők tornyosodnak az égen, akkor önkezebe veszi a sorsát, és az „OLVASHATATLAN KONYV” kifejezést adja tovább. A baj mértéke ez esetben a második paraméterben érkezik. Ha semmi, akkor megnyugodhatunk, ha bármi más, akkor annyira azért nem.

Adattagok

És hiány, és hiány, és végtelen üresség!

Metódusok

Persze, felfoghatjuk optimistán, a latinok ezt tabula rasa-nak nevezték, üres, fehér lapnak, amire bármit lehet írni. Ám nem most, most nem kellenek metódusok!

6. feladat Konyvespolc (18 pont)

Konstruktorok

Paraméteres konstruktor (1 pont): és elérkeztünk arra a pontra, amikor a könyveinket méltó piedesztálra emelhetjük! Nyissunk könyvespolcot! Egy egyszerű leképezés legyen gyűjtője a könyvespolcra kerülő könyveknek, melyet tetszőleges módon létrehozhatunk, csak utána szorgosan töltsük fel a paraméterben kapott leképezés kulcs-érték-párjaival. A lustaság ez alkalommal is az ördög csábítása! Tisztességgel készítsünk másolatot az adatokról!

Adattagok

És itt van adattag, van lágy kenyér és minden földi jószág! Remélhetőleg jól deklarálva. (1 pont)

Metódusok

konyvVasarlas (3 pont): a legvidámabb tevékenységek egyike, olvasni szerető ember számára legalábbis feltétlenül, a könyvvásárlás szent szertartása. Ám rendet muszáj tartani, hogy könyvespolcunk ne süllyedjen káoszba. Előbb is legyenek tehát leltári számok, rövid szöveg formájában, ezeket rendeljük rendre az új könyveinkhez, és az így kapott párokat vizsgáljuk meg a következő szempontok szerint. Egy bizonyos leltári számból csak egy lehet minden könyvespolcon, tehát ha már szerepel nálunk, akkor a könyvet, legnagyobb sajnálatunkra, nem tudjuk felvenni a könyveink közé. Ha az érkező új kötetek között hiányt tapasztalunk, akkor azon lépünk át, az adott leltári számmal együtt. Ám ha ezen zavaró tényezők egyike sem áll fenn, akkor a könyv a mienk, és az új leltári számmal felkerülhet a polcra. A metódus visszaadott értéke azon könyvek száma, amik végül valóban megvásároltunk.

konyvCsere (3 pont): adni jó, kapni jó, cseréljünk néha könyveket! Ha a paraméterben kapott leltári szám létezik kis könyvespolcunkon, és a kapott könyv címe különbözik az általunk ezen a leltári számon tartott könyv címétől, akkor cseréljük le bátran, és ezt egy visszaadott igaz értékkel jelezzük is. Ha a csillagok állása nem kedvező, és a fenti feltételeken elbuktunk, akkor nem tehetünk semmit, csak visszaadhatunk egy hamis értéket.

konyvKolcsonzes (2 pont): bonyolult, ám hasznos kis metódusról van szó! Egy olvasást kap paraméterben, aminek meghívja az olvasasiIdo metódusát, meghozzá a két következő paraméterrel. Elsőnek a könyv fejezeteinek a számát adja, másodjára magát a könyvet. A következő lépés pedig a visszatérési érték vizsgálata. Ha kisebb vagy egyenlő mint 100, akkor minden rendben, ez lenne a visszaadandó érték. Ellenkező esetben segítségért kiált, TulGyorsOlvasas kivételt dob, "a kolcsonzesi ido tul rovid" üzenettel, és második paraméternek adja a könyv bölcséleti értékét, a matematika szabályai szerint kerekítve.

leghosszabbFejezetek (3 pont): tenéked magyarázzam, aki már az utolsó részfeladat közelében jársz? Ez a metódus megvizsgálja az összes könyvet a polcon, megkeresi mindnek a leghosszabb fejezetét, akár segéderőket szólítva maga mellé, melyek természetét nem részletezem, majd ezen leghosszabb fejezetek hosszának összegével tér vissza.

leltariKodok (5 pont): és az utolsó, kissé talán unalmas pont, az adminisztráció, ami minden munkának leglélekölőbb része. Hozz létre ezen metódussal egy „leltar.csv” nevű fájlt, csak úgy, magában, nem részletezett helyen, és írasd bele a könyvek leltári számait, majd pontosvesszővel elválasztva, ám szóköz nélkül a szerző becses neve következzen, és végül újabb pontosvessző után maga a könyv címe. Ezen kiíratás essék meg minden könyvvel a könyvek közül, minden könyv kapjon méltó módon új sort, és minden sor végén ékeskedjék pontosvessző! Ha pedig ez a leírás így magában kevés, egy példával élénk. Legyen három könyvünk, rendre A2, B5,

C7, leltári számokkal, a kiváló Rejtő Jenőtől, a szintén korszakalkotó Szerb Antaltól, és hogy ne csak kis hazánk nagyjait emlegessük, vegyük még mellé a nagyszerű Eric Kastnert is. Ez esetben a fájl tartalma: A2;Rejto Jeno;A tizenegy karatos auto; B5;Szerb Antal;A Pendragon legenda; C7;Eric Kastner;Drei Manner im Schnee;

Hát, ennyit akartam mondani.
Jó munkát!