

Programozás I. 3. zh

SZTE Szoftverfejlesztés Tanszék

2024. tavasz

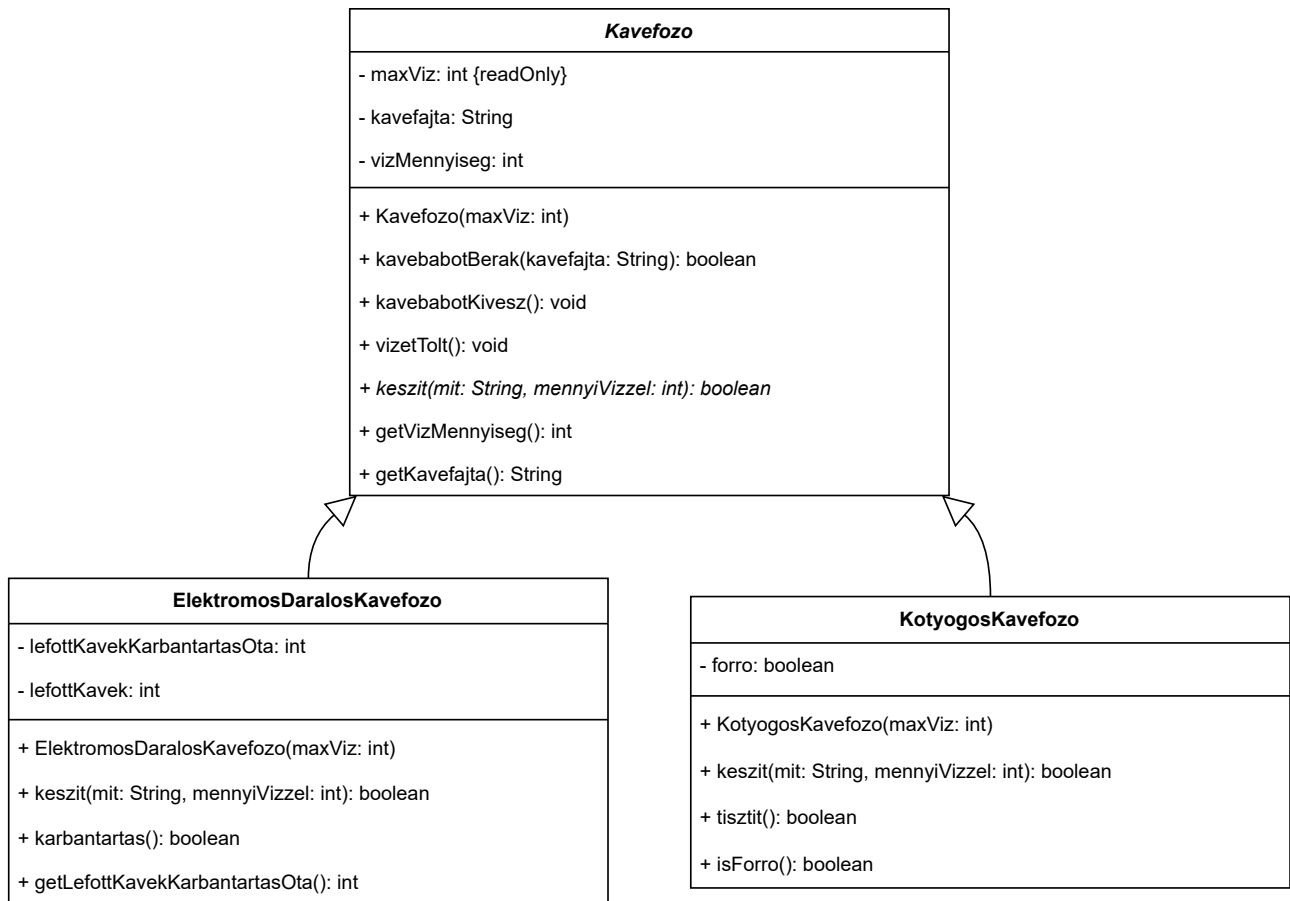
Általános követelmények, tudnivalók

- A feladat elkészítésére 30 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 ZH nem teljesített. Ha valakinek a padtársa segít, akkor mérlegelés nélkül mindkettő hallgató dolgozata sikertelen, a ZH 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.
- A Java elnevezési 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 bemutatott 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 a 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



Az alvás a gyengék kávéja. A feladat megoldása során használd a `minta.zip` fájlban lévő kiindulási projektet, amelyben megtalálható egy absztrakt osztály!

Kavefozo osztály

Az osztály egy tetszőleges kávéfőzőt reprezentál, ahogy az osztálydiagramon is látható. Az osztályban található elkészített elemek:

- **maxViz**: a kávéfőző maximális vízkapacitása, ezt a konstruktor állítja be, és nem változtatható később, jelöld kulcsszóval!
- **kavefajta**: a kávéfőzőben lévő kávéfajta, kezdetben *null*, és null akkor, ha nincs benne kávé.
- **vizMennyiseg**: a kávéfőzőben lévő aktuális víz mennyisége, kezdetben *0*,

Az osztályban lévő metódusok:

- **kavebabotBerak**: kávé berakása a kávéfőzőbe, igazzal tér vissza, ha a művelet sikeres
- **kavebabotKivesz**: kávé kivétele a kávéfőzőből,

- **vizetTolt**: az aktuális vízszintet feltölti a maximumig,
- **keszit**: a kávéfőzést reprezentáló absztrakt metódus,
- **getVizMennyiseg**: vizMennyiseg gettere
- **getKavefajta**: kavefajta gettere

Ezekhez a létező adattagokhoz, metódusokhoz tetszőleges új adattag/metódus hozzáadható, azonban a létező adattagok/metódusok módosítása (láthatóság, típus, egyéb módosítók, működés) nem engedélyezett (hibát dobhat a tesztelő).

A feladat két leszármazott osztály elkészítése: **KotyogosKavefozo**, **ElektromosDaralosKavefozo**.

KotyogosKavefozo osztály (5 pont)

A kotyogós kávéfőzők speciális kávéfőzők, a szükséges új adattag az osztálydiagramon szerepel. Ráadásul most egy különös kotyogósról van szó, hiszen kávébabot kell belerakni.

A **keszit** metódus működése:

- A kotyogós kávéfőző egyszerre 1 típusú kávéval dolgozik, ez az első paramétere. És a teljes benne lévő vízmennyiséget el kell használni, tehát a kotyogós esetében a maximális vízmennyiséget. Értelemszerűen víz nélkül nem lehet kávé készíteni, mert szétég.
- Használat előtt meg kell tölteni vízzel, ehhez használd a *vizetTolt* metódust a *keszit* metódus meghívása előtt!
- Ha a *mennyiVizzel* paraméter értéke nem egyezik a vízmennyiséggel, térjen vissza false-szal!
- Ha nem tud kávé berakni a *kavebabotBerak* metódussal, térjen vissza false-szal!
- A vízszintjét csökkentse le a megfelelő mennyiségre!
- Legyen forró.
- Ha itt minden teljesült, megfőtt a kávé, térjen vissza true-val!

A **tisztit** metódus működése:

- Az elkészítésbe nem tartozik bele a tisztítás, az egy külön procedúra.
- A tisztítás akkor szükséges, hogyha a kávéfőzőben van kávébab. Ha nem szükséges, térjen vissza false-szal.
- Ha van kávébab, akkor a kávébabot vegye ki a *kavebabotKivesz* metódussal!
- A kávéfőző a tisztítás után ne legyen forró!
- Térjen vissza igazzal!

A **isForro** metódus működése:

- Visszaadja a *forro* adattag értékét.

ElektromosDaralosKavefozo osztály (5 pont)

Az elektromos darálás kávézőfőzők speciális kávéfőzők, a szükséges új adattagok az osztálydiagramon szerepelnek.

A **keszit** metódus működése:

- A kotyogós kávéfőző azzal a kávéval dolgozik, ami benne van. Ezt ellenőrizd le!
- Ellenőrizze le, hogy az első paraméterben kapott kávé megegyezik-e a benne lévővel, ha nem, térjen vissza false-szal!
- A vízmennyiséget a fáradt, kávéra szomjazó, esetlegesen függő ember maga választja meg egyéni ízlése szerint. De ha ez több, mint a főzőben lévő víz, akkor térjen vissza false-szal!
- 12 kávéként karbantartani kell, amíg ez nem történt meg, addig térjen vissza false-szal!
- Csökkentse a vízszintet a megfelelő mennyiséggel. És növelje a *lefottKavekKarbantartasOta* változó számát 1-gyel, hogy tudjuk, mikor kell karbantartani!
- Növelje a *lefottKavek* számát 1-gyel!
- 24 kávéként a kávéfőzőből elfogy a kávé, ezt valósítsd meg a *kavebabotKivesz* metódussal!
- Ha elkészült a kávé, térjen vissza true-val!

A **karbantartas** metódus működése:

- Ha nem főztek kávé az utolsó karbantartás óta, térjen vissza false-szal!
- Ellenkező esetben nullázza a *lefottKavekKarbantartasOta* változót, és térjen vissza igazsal!

A **getLefottKavekKarbantartasOta** metódus működése:

- Visszaadja a *lefottKavekKarbantartasOta* adattag értékét.

Jó munkát!