Programozás II. Gyakorló Feladat

SZTE Szoftverfejlesztés Tanszék

2023. ősz

Ismertető

- A programot C++ nyelven kell megírni.
- **A benyújtandó fájl neve kötelezően `feladat.cpp`.**
- A megoldást a *Bíró* fogja kiértékelni.
  - A Feladat beadása felületen a Feltöltés gomb megnyomása után ki kell várni, amíg lefut a kiértékelés. **Kiértékelés közben nem szabad az oldalt frissíteni vagy a Feltöltés gombot újból megnyomni** különben feltöltési lehetőség veszik el!
- Feltöltés után a *Bíró* a programot `g++` fordítóval és a
  `-std=c++1y -static -O2 -DTEST_BIRO=1`
  paraméterezéssel fordítja és különböző tesztesetekre futtatja.
- A program működése akkor helyes, ha a tesztesetek futása nem tart tovább 5 másodpercnél és hiba nélkül (0 hibakóddal) fejeződik be, valamint a program működése a feladatkiírásnak megfelelő.
- A *Bíró* által a `riport.txt`-ben visszaadott lehetséges hibakódok:
  - Futási hiba 6: Memória- vagy időkorlát túllépés.
  - Futási hiba 8: Lebegőpontos hiba, például nullával való osztás.
  - Futási hiba 11: Memória-hozzáférési probléma, pl. tömb-túlindexelés, null pointer használat.
- 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://biro2.inf.u-szeged.hu/Hallg/IBL302g-1/1/hXXXXXX/4/riport.txt`
  3. Az url-ből visszatörölve a 4-esig (`riport.txt` törlése) megkaphatók a 4-es próbálkozás adatai
- A programot 20 alkalommal lehet benyújtani, a megadott határidőig.
- A programban szerepelhet `main` függvény, amely a pontszámításkor nem lesz figyelembe véve. Azonban ha fordítási hibát okozó kód van benne az egész feladatsor 0 pontos lesz.
- A megvalósított függvények semmit se írjanak ki a standard outputra!

Feladatsor

1. feladat (3 pont)

Készíts egy Etel nevű **struktúrát**! Két **publikus** adattagja van: egy sztring, ami az étel nevét tárolja (`nev`) és egy egész szám, ami az étel hűtési igényét tartalmazza (`hutes`).

Rendelkezzen az alábbi konstruktorokkal:

- Legyen egy paraméteres konstruktora, melynek első paramétere egy string, a második egy int és beállítja a két adattagot.
- Legyen egy default konstruktora, amely a sztringet üresre, az int-et 0-ra állítja be

2. feladat (2 pont)

Készíts egy `string print() const;` publikus metódust az Etel struktúrához. A visszaadott sztring tartalma az alábbi legyen: "nev:<nev>,hofok:<hutes>". A <nev> és a <hutes> helyettesítődjenek az adattagokkal, a vessző után **nincs** szóköz.

3. feladat (6 pont)

Készítsd el a program hibaosztályait! Minden osztályban az adattagok, ha vannak, legyenek privát láthatóságúak. A metódusok public-ok legyenek.

- class **Hiba**. Publikusan öröklődik az `std::exception` osztályból. Tulajdonságai:
  - Felüledfiniálja az ős `what` metódusát, hogy ezt a sztringet adja vissza: "Hiba tortent".
- class **Megromlik**. Publikusan öröklődik a Hiba ősosztályból. Akkor kell dobni, ha az ételt rossz hőfokon tároljuk le. Az alábbiakkal rendelkezik:
  - Egy konstruktor, amely egy Etel-re mutató konstans referenciát vár paraméterül
  - Felüldefiniálja az őse `what()` metódusát úgy, hogy az alábbi sztringet adja vissza:
    - "A <etel.nev> tarolasahoz legalabb <etel.hutes> fok kell"

        * Ahol a kacsacsőrök közötti részek helyettesítődnek a konstruktorban érkezett étel megfelelő adattagjaival

- class **Megtelt**. Publikusan öröklődik a Hiba ősosztályból. Az alábbiakkal rendelkezik:
  - Egy konstruktor, amely egy std::string-et vár paraméterül
  - Felüldefiniálja az őse `what()` metódusát úgy, hogy az alábbi sztringet adja vissza:
    * "Nem sikerult letarolni a <mit>-t"
    * Ahol a kacsacsőrök közötti rész helyettesítődik azzal a sztringgel, ami a konstruktorban érkezett

4. feladat (4 pont)

Készíts egy Hutoegyseg nevű osztályt! Egy hűtőegység rendelkezik hűtési hőfokkal (egész szám), emellett 10 db étel eltárolására képes. A tárolt ételeket tartalmazó tömböt lehessen lekérdezni a `get_tartalom` getterrel, míg a hűtési hőmérsékletet a `get_hofok` metódus adja vissza.

Készítsd el a Hutoegyseg osztály konstruktorát is! Egy **string** paramétert vár, amely a hűtőegység hűtési hőfokát reprezentálja sztringes formában. Konvertáld intté a kapott értéket, de figyelj oda rá, hogy hiba esetén a konstruktor ne dobjon kivételt, hanem állítsa be -1-re a hőfokot! A hűtőegység kezdetben nem tárol ételt.

5. feladat (5 pont)

Valósítsd meg a += operátort a Hutoegyseg osztályban. Paramétere egy Etel legyen! Az operátor feladata, hogy hozzáadja a paraméterben kapott ételt a tartalom tömbhöz. Legyen benne hibakezelés is! Először a hőmérséklet megfelelősége legyen vizsgálva. Az étel csak akkor tehető be a hűtőegységbe, ha hűtésigénye nagyobb vagy egyenlő, mint a hűtőegység által biztosított hőfok. Ha ez nem teljesül, akkor legyen egy Megromlik kivétel dobva, mely kivétel a paraméterben kapott étellel van inicializálva. Ha a hőfok feltétel teljesül, akkor az legyen ellenőrizve másodiknak, hogy van-e még hely a tömbben. Ha nincs, akkor legyen egy Megtelt kivétel dobva az étel nevével inicializálva. Hiba esetén ne legyen a hűtőegységhez étel adva.

6. feladat (3 pont)

Készíts egy `print()` publikus metódust a Hutoegyseg osztályhoz. Feladata, hogy végigmenjen az aktuálisan tárolt ételeken és a standard outputra kiírja, amit az ételek `print()` metódusa visszaad. Az egyes print hívások után legyen mindig egy-egy sortörés is a standard outputra írva. Legyen Hiba típusú kivétel dobva, ha a hűtőegységben nincs étel tárolva.

7. feladat (4 pont)

Készíts egy globális (osztályokon kívüli)
`bool feltolt(Hutoegyseg& h, Etel etelek[], unsigned etelszam)`
függvényt, amelynek feladata, hogy feltöltse a hűtőt a += operátor meghívogatásával. Menjen végig a függvény a paraméterben kapott tömb elemein. A hosszt a harmadik paraméter adja

meg. Mivel a Hutoegyseg interfészéből nem kérdezhető le a kapacitás, így előzetes méretellenőrzést nem kell/nem lehet belerakni. Fontos, hogy a `feltolt` függvényből ne dobódjon kivétel. Ha hiba történne fejeződjön be a hűtő feltöltése és legyen false érték visszaadva. Ha minden étel letárolása sikerült, akkor legyen true visszaadva.