

3. Beadandó feladat dokumentáció

Készítette:

Giachetta Roberto

E-mail: groberto@inf.elte.hu

Feladat: Készítsünk programot, amely Marika néni kávézójának eladási nyilvántartását végigköveti.

- A kávézóban 3 féle étel (hamburger, ufo, palacsinta 10%-os áfával), illetve 3 féle ital (tea, narancslé, kóla 25%-os áfával) közül lehet választani.
- Az ételek ezen belül különfélék lehetnek, amelyre egyenként lehet árat szabni, és elnevezni (az elnevezés része a fajtája is, ezt nem kell megadni), az italok árai rögzítettek (bruttó 200 Ft, 300 Ft, 250 Ft).
- Minden rendelés több ételből és több italból áll, és sorszámmal rendelkezik, esetlegesen törzsvásárlói számmal, amelyet hagyhatunk üresen is.
- Lehessen lekérdezni egy adott rendelés részleteit, valamint egy adott törzsvásárló összes rendelésének részleteit, illetve összesítve a bruttó, nettó összeget a teljes eladásra, és törzsvásárlóra.

Elemzés:

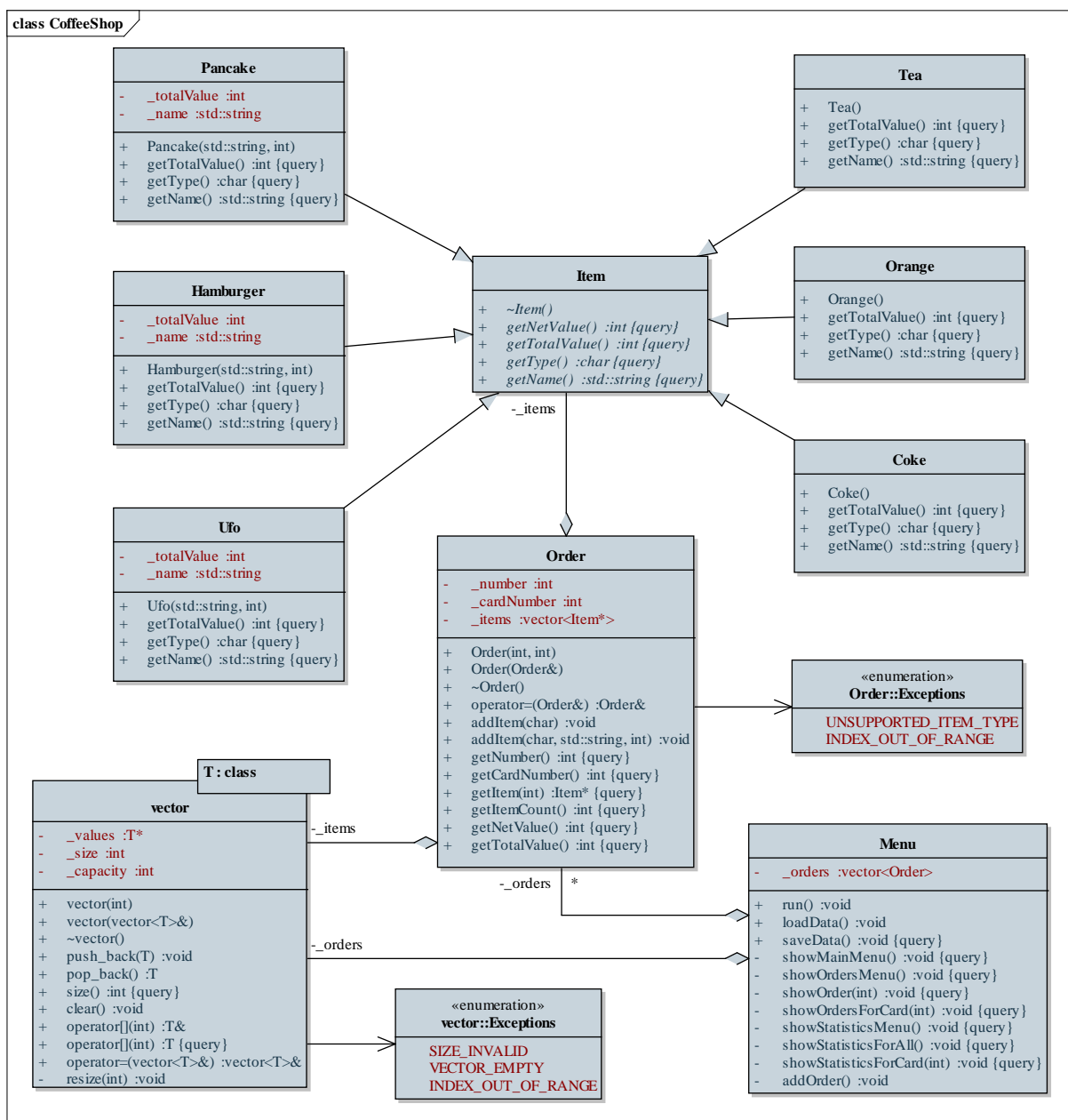
- A program egy menün keresztül vezérelhető, amelyben elérhető új rendelés bevitele, a bruttó/nettó összeg megtekintése egy adott rendelésszámmra, egy adott törzsvásárlói számmra, vagy összesítve.
- Új rendelés bevitelénél megadhatóak az ételek, az italok, valamint a törzsvásárlói kártya száma. Az italokat csak ki kell választani, ételek esetén megadható a pontos megnevezés, illetve ár.
- Az adatokat egy szöveges fájlban (kavezo.dat) tároljuk, amelyből az adatok a program megnyitáskor betöltődnek, bezárásakor pedig elmentődnek.
- A fájl szerkezete:

```
<rendelés azonosító>;<italok/ételek száma>;<törzsvásárlói szám>  
<típus: h/u/p/t/n/k>;<étel neve>;<étel ára>  
<típus: h/u/p/t/n/k>;<étel neve>;<étel ára>  
...  
<rendelés azonosító>;<italok/ételek száma>;<törzsvásárlói szám>  
...
```

Tervezés:

- A program vezérlését egy menü (**Menu**) biztosítja. A rendeléseket a menüben vektor segítségével tároljuk. A menüben található az adatok betöltése (**loadData**), és mentése (**saveData**), továbbá az egyes menüfunkciókat megvalósító, illetve kiíró műveletek, amelyek a menü futtatása (**run**) fog össze.

- A rendeléshez felveendő tételek szükségünk van a hat különböző tételtípusra, amelyek közös tulajdonságai a típusnév, a bruttó és nettó ár, e mentén általánosíthatunk, létrehozva a tétel (**Item**) absztrakt osztályt. Az ételek további tulajdonságai a név és ár, amelyeket mezők segítségével tároljuk el az egyes osztályokban. Biztosítjuk a lekérdező műveletek felüldefiniálhatóságát, ezek közül csak a nettó értékre adunk alapértelmezést a tételben (a többi tehát absztrakt lesz).
- A rendelések (**Order**) mezői az azonosító (ez automatikus), a törzsvásárlói szám, valamint a tételek, amelyeket polimorfizmus segítségével egy közös vektorban tárolunk. Továbbá lehetőséget biztosítunk tétel hozzáadására (külön italokra, illetve ételekre, utóbbiaknál a nevet és árat is meg kell adni).



A program osztálydiagramja

- Tetszőlegesen sok elem fogadásához a rendelésben, valamint a menüben egy olyan sablonos vektort (**vector**) használunk, amelyet dinamikus tömbbel ábrázolunk, és amely automatikusan méreteződik a tartalomhoz.

Implementáció:

- A megvalósításban a következő modulokra bontjuk a programot: *menu*, *order*, *item*, *vector*.
- A lekérdezésekhez külön összesítünk összegzés tételével a rendelésben, valamint a menüben. A nettó értékeket a bruttó értékekből számoljuk minden rendeléselemnél.
- Mivel a rendelési tételt polimorfizmussal kezeljük, a rendelés mutatók vektorát tartalmazza, emiatt a rendelésre meg kell valósítani a másoló konstruktor, értékadás és destruktorkonstruktor műveleteket.
- Hasonlóan, mivel a vektor dinamikus tömbbel épül fel, szintén meg kell valósítani a másoló konstruktor, értékadás és destruktorkonstruktor műveleteket.
- A vektor esetén kivételekkel jelezzük az esetleges hibákat. A lehetséges hibák: rossz kezdeti méret (**SIZE_INVALID**), kivétel üres vektorból (**VECTOR_EMPTY**), valamint hibás indexelés (**INDEX_OUT_OF_RANGE**). A rendelés esetén is felveszünk két kivételt: hibás indexelés (**INDEX_OUT_OF_RANGE**), valamint nem megfelelő tételtípus (**UNSUPPORTED_ITEM_TYPE**).
- A fájl betöltését és mentését soronkénti feldolgozással végezzük, a megadott szerkezetnek megfelelően.