***Számítógép konfiguráció összeállító rendszer***

***Kovács Máté C8N5NZ***

**A .txt fájl leírása:**

A text fájlban egy sorban van felsorolva minden típus a ’#’ karakter elválasztásával, majd a ’\*’ karakterrel egy adott eszköz aminek a különböző tulajdonságait pedig a ’,’ karakter választja el egymástól. (bin\debug mappába kell rakni)

**IHardverElem interfész:**

az interfész tartalmaz 4 különböző tulajdonságot és egy enum-ot is, amikkel majd az eszközeinket létrehozzuk. Illetve két féle metódust, a beépít és az elromlik, ezeket az interfészt megvalósító osztályok fogják majd implementálni.

**Szamitogep osztály:**

a számítógép osztály rendelkezik 4 darab mezővel, ezeken foglyuk tárolni a különböző eszközöket amikor egy gépet összerakunk, ez lesz a beépít metódus. Ebből annyi példányt hozunk létre a main metódusban amennyi számítógépet összeakarunk rakni.

**Alaplap, ram, videokartya osztályok:**

ezek az osztályok megvalósítják az IHardverElem interfészt és annak tulajdonságainak értéket adnak a konstruktoraikon keresztül. A beépít metódust implementálva mindegyik paraméterként egy Szamitogep objektumot vár, amibe majd az adott eszköz beépítésre kerül. Megnézi, hogy a paraméterként átadott objektumban található-e már az adott osztály típusa, ha talált kivételt dob, ha nem talált akkor beépíti – azaz a szamitogep objektum azonos típusú mezőjébe tárolja. Az Elromlik metódus meghívásakor minden esettben kiírja a képernyőre a meghibásodott eszköz nevét és típusát.

**Processzor és IntelProcesszor osztály:**

A Processzor osztály egy absztrakt osztály, ő valósítja meg az IHardverElem interfészt. Hasonlóan, mint az előző osztályokhoz, ő is értéket ad a konstruktorán a tulajdonságoknak, illetve beépít és elromlik metódusa is ugyan úgy működik. Az IntelProcesszor ellenben a Processzor osztály leszármazottja, ő meghívja az őse konstruktorát és azon keresztül kapja meg az értékeket.

**Graf osztály:**

A gráf osztály egy generikus osztály. Szomszédsági listával van megvalósítva. Van neki egy tartalom nevű generikus listája, amiben a csúcspontokat tároljuk – azaz egy adott eszközt, illetve egy szomszed nevű generikus listát tartalmazó lista, amiben pedig a csúcspontok szomszédjait tároljuk. Az AddNode metódus hozzáad egy új csúcspontot az adatszerkezethez. Az AddEdge pedig megkeresi a paraméterként átadott csúcspontok helyét(indexét) amik között szeretnénk kapcsolatot és a szomszédjaik közé beszúr egy utat. A Neighbors metódus a paraméterként bekért csúcspontnak megkeresi az indexét és az azon indexű szomszédjait adja vissza. A KonkretKereses metódus egy string paraméterrel rendelkezik, arra alkalmas, hogy ha egy konkrét nevű eszközt szeretnénk és a vele kompatibilis eszközöket, akkor a paraméterként átadott értéket megkeresi a tartalom nevű listánkban és amennyiben megtalálta azt, az indexét eltároljuk és visszaadjuk az azon index szomszédjait. A TipuKereses hasonlóan működik, mint a KonkretKereses, csak itt arra vagyunk kíváncsiak milyen eszközökből választhatunk egy adott típusból. Átadjuk neki milyen típust keresünk, majd a tartalom nevű listában megkeresi az azonos típusú elemeket és kiírja a képernyőre a nevét, árát és minőségét. A Legolcsobb és LegjobbMinoseg metódus mindkét esetben egy számítógép objektumot add vissza, ami vagy a legolcsóbb összeállítást tartalmazza vagy a legjobb minőségűt. A paraméter ezúttal egy gráf. A legolcsóbb összeállításnál előszőr megkeresi a legolcsóbb alaplapot a tartalom nevű listában, amint megtalálta eltároljuk az árának értékét és az indexét, ahol megtalálta. Ezután a szomszéd listában keresünk tovább méghozzá azon az indexen, ahol megtaláltuk a legolcsóbb alaplapot, megnézzük, hogy melyik a legolcsóbb processzor, ram és videókártya, amivel találunk közös éleket, ha megvannak eltároljuk egy számítógép objektum tulajdonságaiban és ezt az objektumot adjuk visszatérési értékként. A legjobb minőség is ezen az elven alapszik, csak ott a minőség tulajdonságot hasonlítjuk és a legnagyobb értéket keressük. Szintén megkeresi a legjobb minőségű alaplapot a tartalom listából, majd ennek a szomszédjait nézi meg és választja ki a legjobb minőségű eszközöket. A visszatérési érték itt is egy felszerelt számítógép objektum lesz.

**Main metódus:**

Itt történik a fájlbeolvasás, illetve létrehozzuk a gráfunkat. A gráfunk IHardverElem típusú elemeket képes tárolni. Itt tudunk létrehozni számítógép objektumokat is, amiket később felszerelünk eszközökkel. A beépít és elromlik metódusok is meghívásra kerülnek ebben a részben. A fájlbeolvasás után a különböző eszközöket a saját osztály típusú tömbjükben érjük el (pl. alaplapokat – alaplap[i]). A gráfunk csúcspontjai és élei is itt valósulnak meg. Mindkét összeállítás és a keresések eredménye is itt kerül kiíratásra a képernyőre.