

2. ZH

GYAKORLÁS

Készítse el az alábbi alkalmazást!

Az Apple nemrég adta ki az iPhone legújabb operációs rendszer frissítését (iOS 10.1.1), ami több felhasználó szerint programhiba miatt rosszul értelmezi az akkumulátor töltöttségi szintjét, így idő előtt kikapcsol a telefon. A helyzetet bonyolítja, hogy a probléma csak bizonyos modellek esetén fordulhat elő: iPhone 5, iPhone 6, iPhone 6s, iPhone 6s Plus.

Készítsen alkalmazást, ami képes analízálni a fájlból (**MONITORING.DAT**) beolvasott akkumulátor töltöttségi szinteket, majd fájlba írni az egyes készülékeknél tapasztalható hiba okát (OS - programhiba vagy AC – akkumulátor hiba vagy OK).

A fájl első sora a telefonok számát tartalmazza, majd a további **N** sor, soronként kettősponttal (:) elválasztva :

- a telefon azonosítóját,
- a telefon típusát,
- az akkumulátor töltöttségi szintjeit (a töltöttségi értékek pontosvesszővel (;) vannak elválasztva).

5	
8940560:iPhone 5:90;88;70;67; 30;4 ;44;47	8940560:OS
0930840:iPhone 6s:89;84; 80;72 ;70;66;64;53;48;40;31	0930840:AC
6840556:iPhone 6:60;40; 30;2 ;5;7;9;11;14	6840556:OS
9870650:iPhone 6s Plus:55;50;51;53;57; 51;37 ;40;44;47	9870650:AC
0498355:iPhone 5:60;55;50;45;40;35;30;25;30;33;37	0498355:OK

Az analízis a következő szempontok alapján dönti el a hiba valódi okát (az előrébb lévőket kell figyelembe venni):

- Ha a 30-as érték szerepel a mért értékek között, amit 5-nél kisebb érték követ, akkor: OS.
- Ha van olyan csökkenő sorozat, amiben több mint 7 egység a különbség az értékek között, akkor: AC.
- Minden más esetben a telefon megfelelően működik, vagyis: OK.

Az alkalmazás indulását követően ne kérjen be adatot a felhasználótól és ne írjon ki számára semmit. A beolvasandó állomány a futtatandó .exe mellett szerepel MONITORING.DAT névvel, ahova a kimeneti állományt kell elkészíteni ANALYSIS.RESULT névvel. Az alkalmazás teszteléséhez készítsen egy MONITORING.DAT állományt. **2 pont**

Az alábbi felépítést követve készítsen osztályokat, és valósítsa meg a leírás alapján az analízist. Ügyeljen rá, hogy betartsa az objektum-orientált programozásban használatos egységbezárási és adatretjezési elveket. Ahol szükséges az adatmezőkhöz hozzon létre tulajdonságokat, ha elegendő csak olvashatót, illetve a példányhoz nem kötődő metódusokat, és adatmezőket tegye statikussá.

2 pont

FaultCause felsorolás:

3 pont

- A hiba okainak rövidítéseit tartalmazza: OS, AC, OK.

Phone osztály:

12 pont

- Tulajdonságok, melyek az osztályon kívül csak olvashatók: ID, Type, BatteryLevels.
- Tulajdonság, mely az osztályon kívül írható és olvasható is (a hiba okát tartalmazza): FC.
- Konstruktor, ami megkapja a telefon adatait: id, type, batteryLevels, ami alapján beállítja a tulajdonságok értékeit, illetve az FC értékét alapértékként OK-ra állítja.
- ToString() metódus, ami visszaadja az osztály ID és FC tulajdonság értékét kettősponttal (:) elválasztva szöveggént.
- (Az osztály az akkumulátor töltöttségi szintjeit egy számtöbbsen tárolja.)

FileHandler (static) osztály:

13 pont

- Osztályszintű tulajdonságok, melyekben az útvonalak vannak eltárolva: srcPath, dstPath.
- Osztályszintű Load() metódus, ami a forrás útvonalról beolvassa a fájl tartalmát, majd soronként bontva egy szöveg tömbben visszaadja.
- Osztályszintű Append() metódus, ami a cél útvonalon lévő fájlhoz hozzáfűzi a paraméterül kapott szöveges változó tartalmát.

Analyzer osztály:

14 pont

- Adatmező, mely a telefon példányokat tartalmazza: phones[].
- Konstruktor, ami a FileHandler segítségével beolvassa a fájl tartalmát, majd létrehozza a Phones példányokat, amit a phones tömbben tárol el.

- `Analysis()` metódus, ami analizálja a `phones` tömb objektumainak akkumulátor töltöttségi szintjeit, majd az egyes telefon hibákat az analízist követően rögtön hozzáfűzi a kimeneti fájlhoz. A prioritás szerint hívja meg előbb az `OSCause()` metódust, majd ha nem emiatt történt a hiba, akkor az `ACCause()` metódust.
- `OSCause()` metódus, ami azt vizsgálja, hogy a hibát az operációs rendszer okozta-e. Ha igen, akkor beállítja a vizsgált példány `FC` tulajdonságát, majd igaz értékkel visszatér a metódus. Bementi paramétere a metódusnak a vizsgálandó `phone` példány.
- `ACCause()`, metódus, ami azt vizsgálja, hogy a hibát az akkumulátor okozta-e. Ha igen, akkor beállítja a vizsgált példány `FC` tulajdonságát, majd igaz értékkel visszatér a metódus. Bementi paramétere a vizsgálandó `phone` példány.

Program osztály:

4 pont

- `Main()` metódus: A `FileHandler` osztály megfelelő tulajdonságainak értékül adja a fájl útvonalakat, majd példányosítja az `Analyzer` osztályt, végül meghívja az `Analysis()` metódusát.

Értékelés:	1: 0 – 24,5p	2: 25 – 30,5p	3: 31 – 36,5p	4: 37 – 42,5p	5: 43 – 50p
------------	---------------------	----------------------	----------------------	----------------------	--------------------