

Készítsünk egy osztályt, amelynek konstruktora a paraméterként kapott, double típusú tömböt (pointert) eltárolja az adattagjában (méretet is tárolni!). Írjunk egy paraméter nélküli metódust, amely az elemek átlagával tér vissza. Teszteléshez a tömb elemeit vegyük text fájlból (ha átkapcsoltunk hu-hu-ra, tizedesvessző), egy sorban valahány, változó számú adattal, tab-bal szeparálva. Egy sor feldolgozása után az osztály példányát töröljük, és a következő sor már a következő példányba kerüljön.

Készítsünk egy osztályt, amely megkapja két ember teljes nevét (vagyis két stringet), majd egy metódusa megállapítja, hogy azonos-e a keresztnévük, ez lesz a visszatérő értéke (ha több keresztnév is van, akkor tekintjük azonosnak, ha az összes keresztnév azonos). Az osztályt két input string segítségével teszteljük is le, amelyeket egy text fájlból veszünk: soronként 2 teljes név szerepeljen, tabulátorral elválasztva

Készítsünk objektum osztályt, amely konstruktorában megkap egy kétoperandusú számtani alpműveletet stringként, a számok közt a műveleti jellel (pl. „12+34”). Metódussal végezzük el a műveletet, amelynek visszatérő értéke a művelet eredménye legyen. Nem feltétlenül kétjegyű egész számokkal dolgozunk! Vegyük észre a tizedespontot/vesszőt, az objektum tartalmazzon egy bool is_int() nevű metódust, amely igaz, ha egész műveletet végzünk, hamis, ha valósat. Ennek megfelelően két eredmény metódus kell, külön a valósra és az egészre. Az objektumot példányosítsuk, és egy szövegfájl soraiban található tesztadatok segítségével teszteljük is le. Miért nem tudunk overload-ot alkalmazni?

Készítsünk objektum osztályt, amely konstruktorában kap egy C++ programsort stringként. A nyelv definíciója szerint ha // van a sorban, a sor további része megjegyzés (comment). Készítsünk két, string típusú metódust, amelyekkel az utasításrész és a komment rész külön-külön kiszedhető a konstruktorban megadott stringből. Nulla hosszúságú sorral is működnie kell, ekkor az utasításrész is és a komment rész is 0 hosszúságú. Az objektumot példányosítsuk, és egy .cpp textfájl soraival (pl. saját magával!) teszteljük is le. Szorgalmi feladat: ha páratlan számú idézőjel van a // előtt, nem számít kommentnek: cout<<"//"<<endl;