Programozás Pythonban 40 pont

* Az Ön feladata az alábbiakban olvasható leírás alapján három program elkészítése. Minden program első sora megjegyzésben tartalmazza a saját nevét!
* A programokat a saját gépén létrehozott VIZSGA mappába kell mentenie.
* A programok elkészítése során a felhasználó által megadott adatok helyességét nem kell ellenőriznie (ha például a program egy 1 és 5 közé eső szám megadását kéri a felhasználótól, akkor feltételezheti, hogy a felhasználó számot, és a megadott feltételeknek megfelelő számot ad meg)
* Az összes képernyőre történő kiíratásnál a minta szerint készítse el a feladatokat, az ékezetmentes kiírás is elfogadott.
* Törekedjen arra, hogy a tanult programozási elveknek megfelelő adatszerkezeteket, vezérlési szerkezeteket alkalmazzon!
* Munkáját rendszeresen mentse! Amennyiben a vizsga során a számítógép nem megfelelő működését tapasztalja, jelezze a felügyelő tanárnak!

1. **feladat: Hosszú szavak** (8 pont)

Írjon programot **szavak.py** néven! A program kérjen be, egy tetszőleges szót, majd tárolja el egy változóban. A szavakat addig be kell kérni amíg, legalább öt karakterből álló szót adunk meg. Amint megfelelő hosszú szót adunk meg a program kiírja „Az ellenőrzött bekérés sikeres!” üzenetet, majd kiírja a megadott szót és hosszát, valamint szót visszafelé. Azokat a részeket, amiket a felhasználó gépel be, a mintában vastagított és döntött betűkkel emeltük ki.

>>> %Run nagyobb.py

Adj meg egy számot! ***12***

Adj meg egy másik számot! ***5***

A nagyobb szám: 12

>>> %Run nagyobb.py

Adj meg egy számot! ***25***

Adj meg egy másik számot! ***63***

A nagyobb szám: 63

>>> %Run nagyobb.py

Adj meg egy számot! ***8***

Adj meg egy másik számot! ***8***

A két szám egyenlő

1. **feladat: Kockák tulajdonságainak kiszámítása** (14 pont)

Írjon programot **kockak.py** néven!

Írjon egy termes nevű függvényt, ami egy paraméterként kapott valós szám (élhossz) alapján kiszámítja az élhosszhoz tartozó kocka térfogatát, majd az így kapott valós szám legyen a visszatérési érték!

(Segítség: kocka térfogata: , ahol „a” a kocka élhossza)

Kérje be a kockák számát (legalább 1), majd a magadott számnak megfelelő számú élhosszat (valós szám) is kérje be, ezt tárolja egy listába és a listát írassa ki a képernyőre!

A számítsuk és írassuk a képernyőre listában tárolt élhosszhoz tartozó kockák jellemzőit: az elkészített függvény alapján a kocka térfogatát és testátlóját.

(Segítség: kocka testátlója: , ahol „a” a kocka élhossza)

A program üzeneteinek megfogalmazásában kövesse az alábbi példát! Azokat a részeket, amiket a felhasználó gépel be, a mintában vastagított és döntött betűkkel emeltük ki.

>>> %Run vizsga.py

Add meg a vizsgázó nevét! ***Próba Piroska***

Add meg a pontszámát! ***52***

Add meg az elérhető maximális pontszámot! ***120***

Próba Piroska vizsgája sikeres.

Add meg a vizsgázó nevét! ***Nagy Viktória***

Add meg a pontszámát! ***28***

Add meg az elérhető maximális pontszámot! ***100***

Nagy Viktória vizsgája sikertelen.

Add meg a vizsgázó nevét! ***Molnár Péter***

Add meg a pontszámát! ***40***

Add meg az elérhető maximális pontszámot! ***100***

Molnár Péter vizsgája sikeres.

Add meg a vizsgázó nevét!

>>>

1. **Könyvesbolt készlete (18 pont)**

Az elkészítendő program egy könyvesbolt készletét adatait: a könyv címét, műfaját, árát és készleten lévő könyvek darabszámát tárolja objektumokban.

Ezeket az objektumokat egy konyvek.txt-ben tárolt adatokból töltjük fel. Az állomány utf-8 kódolású adatokat tartalmaz. Az adatokat a ; karakter választja el egymástól. A konyvek.txt egy sora: Gyilkosság az Orient expresszen;krimi;3000;6 A sorokban lévő adatok a következők:

* a könyv címe: pl. Gyilkosság az Orient expresszen
* a könyv stílusa: pl. krimi
* a könyv ára Ft-ban: pl. 3000
* a leltáron lévő mennyiség: pl. 6

A listába tárolt objektumok alapján kimutatásokat készítünk, amely adatokat részben a képernyőn jelenítjük meg, részben a rendeles.txt fájlban tároljuk el.

Írjon programot **konvesbolt.py** néven! Az összes képernyőre történő kiíratásnál a minta szerint készítse el a feladatokat, az ékezetmentes kiírás is elfogadott.

Az adatok tárolására használja, a Konyv osztályt mely részben van elkészítve. Egészítse ki az osztálydefiníciót úgy, hogy az objektumok alkalmasak legyenek a készleten lévő könyvek darabszámának tárolására is!

Olvassa be a konyvek.txt fájlt, darabolja fel a sorokat a ; mentén és tárolja el az adatokat egy olyan listában, ahol egy lista elem egy osztály példánynak felel meg! Például a lista nulladik elemének van címe, típus, stílusa, ára és darabszáma is.

Írassa ki, hogy hány fajta könyv van a konyvek.txt-ben (minden cím egy kölön fajtának számít)!

Írassa ki, hogy összesen hány könyv, illetve hány krimi van a könyvesboltban készleten!

Határozza meg, hogy a könyvtárban melyik a legdrágább könyv! Írassa ki a könyv címét és árát!

Írja ki az utf-8 kódolású rendeles.txt fájlba azokat a könyveket, amely 3-nál kevesebb a készleten levő darabszámát.

A szöveges fájlt a programból kell létrehozni és beleírni! (Amennyiben nem sikerül az adatbekérés és feldolgozás, de szöveges fájlt tud létrehozni, úgy a „Nem sikerült a feladaot megoldani.” szöveget írassa fájlba.)

A program üzeneteinek megfogalmazásában kövesse az alábbi példát!

>>> %Run naplo.py

Diák neve: ***Nagy László***

Súlya(kg): ***76.5***

Magassága(cm): ***181***

Diák neve: ***Kis Lilián***a

Súlya(kg): ***50.2***

Magassága(cm): ***169***

Diák neve: ***Molnár Csaba***

Súlya(kg): ***80.3***

Magassága(cm): ***185***

Tanulónyilvántartás

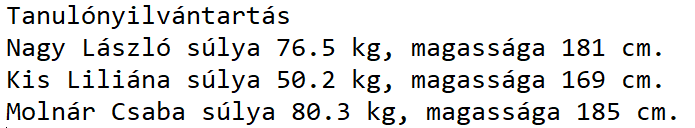
Nagy László súlya 76.5 kg, magassága 181 cm.

Kis Liliána súlya 50.2 kg, magassága 169 cm.

Molnár Csaba súlya 80.3 kg, magassága 185 cm.

>>>

#### A naplo.txt fájl tartalma a fenti adatok megadása esetén:



#### A konyv.py modul tartalma

# a könyv osztály definiálása

class Konyv:

def \_\_init\_\_(self, cim, mufaj, ar, darab):

self.cim = cim

self.mufaj = mufaj

self.ar = ar