1. **Programozás Pythonban (40 pont)**

A feladatok megoldásánál vegye figyelembe az alábbiakat:

* Az egyes feladatokban a képernyőre írást mindig a mintának megfelelően valósítsa meg.
* Amennyiben a felhasználótól kér be adatot, csak akkor kell ellenőriznie a bekért adat helyességét, ha azt a feladat szövege külön kéri, egyéb esetben feltételezheti annak helyességét.
* A 3. feladat megoldásakor az állományban lévő adatok helyes szerkezetét nem kell leellenőriznie, feltételezheti, hogy a rendelkezésre álló adatok a leírtaknak megfelelnek.
* A 3. feladat megoldását úgy készítse el, hogy az azonos szerkezetű, de tetszőleges bemeneti adatok mellett is helyes eredményt adjon!

A három különálló feladat megoldásait a saját mappáján belül található „megoldasok\B-programozas” mappába mentse a megadott nevekkel!

Törekedjen arra, hogy a tanult programozási elveknek megfelelő adatszerkezeteket, vezérlési szerkezeteket alkalmazzon!

**1. feladat 8 pont**

Írjon programot ***derekszogu.py*** néven, amely bekéri egy derékszögű háromszög két befogójának hosszát (egész számok), majd kiszámítja az átfogót.

A kiírást a minta szerint valósítsa meg! Az átfogó hosszát 1 tizedesjeggyel jelenítse meg. Feltételezheti, hogy a felhasználó helyes adatokat ad meg.

1. feladat

Kérem, adja meg az egyik befogót: a = 5

Kérem, adja meg a másik befogót: b = 9

A háromszög átfogója: c = 10.3

**2. feladat 14 pont**

Írjon programot ***hasonlo.py*** néven! Hozza létre a *nagyit(a,x)* nevű függvényt, amely egy négyzetet nagyít fel a megadott aránnyal! A függvény egyik bemenő paramétere a négyzet oldala, a másik a hasonlóság aránya legyen, visszatérési értéke pedig a felnagyított négyzet oldalának a hossza!

A függvény teszteléséhez állítson elő egy egész véletlenszámot a négyzet oldalának hosszára a [10; 20] zárt intervallumból. Írja ki az értéket a mintának megfelelően. Ezután a felhasználótól kérje be a hasonlóság arányát, ami egy 1-nél nagyobb valós szám legyen! A bekérést addig végezze, amíg a felhasználó 1-nél nagyobb számot nem ad meg!

Kövesse a minta szerinti megjelenítést, amely egy lehetséges kimenetet tartalmaz!

Az adatok generálását illetve bekérését a függvényen kívül oldja meg! Feltételezheti, hogy a felhasználó számot ad meg.

2. feladat

Az eredeti négyzet oldala 12 cm.

Kérem, adja meg a nagyítás mértékét (1-nél nagyobb szám): 0.5

Kérem, adja meg a nagyítás mértékét (1-nél nagyobb szám): 2.4

A felnagyított négyzet oldala 28.8 cm.

**3. feladat 18 pont**

Az utf-8 kódolású meghivok.txt állomány a végzősök által igényelt ballagási meghívók számát tartalmazza az alábbiak szerint:

Rém Elek;12A;7

Szőke Mátyás;12A;3  
...

Az állomány egy sora tartalmazza a tanuló nevét, osztályát, valamint az igényelt meghívók számát, pontosvesszővel elválasztva.

Írjon programot ***ballagas.py*** néven! Olvassa be és tárolja el a meghivok.txt állomány tartalmát egy arra alkalmas adatszerkezetben. Amennyiben a beolvasást nem tudja megvalósítani, akkor a felhasználótól kérjen be 5 adatsort!

Az adatok tárolására szolgáló objektumok alapját képező osztály rendelkezésére áll, a meghivok.py állományban. Töltse be ezt a modult (vagy másolja be a ballagas.py fájlba) és használja fel a benne lévő osztályt!

class Meghivo:

def \_\_init\_\_(self, nev, osztaly, darab):

self.nev = nev

self.osztaly = osztaly

self.darab = darab

a) Határozza meg és írja ki a képernyőre, hogy összesen hány adatsor található a forrásállományban!

b) Függvény segítségével határozza meg, összesen hány meghívót igényeltek a diákok!

c) Függvény segítségével határozza meg, melyik diák kérte a legtöbb meghívót! Feltételezheti, hogy csak egy ilyen diák van. A tanuló valamennyi adatát jelenítse meg a mintának megfelelően! Ehhez egészítse ki az osztályt egy olyan függvénnyel, amellyel az adott objektum tartalmát kiíratásra alkalmas formában adja vissza!

3. feladat

a) Az állomány 160 adatsort tartalmaz.

b) A végzős diákok összesen 1437 db meghívót igényeltek.

c) A legtöbb meghívót igénylő tanuló:

tanuló neve: Gipsz Jakab, osztálya: 12P, meghívók száma: 16