

# Saját eredmények

Dolgozatomban első felében javaslatot teszek a Magyarországon használt makroszeizmikus kérdőív átalakítására, melynek három fő célja van:

- könnyebbé váljon a kitöltés, ezáltal növekedjen a beérkező kérdőívek száma,
- jobban automatizálható legyen ezt követően a kiértékelés,
- a kérdések jobban harmonizáljanak az európai EMS-98 intenzitásskálával.

Munkám során összevettem a magyar kérdőívet 9 másik európai ország kérdőívével, illetve a EMS-98 skálával, elvégeztem ezek leíró elemzését és azonosítottam az átvehető jó gyakorlatokat.

Dolgozatom második felében ismertetem a kérdőívek kiértékelésének lehetséges számítógépes módszereit, illetve az elkészített Python nyelvű kiértékelőprogramot, amely a (Musson, 2006) szerinti algoritmust használja.

A program teszteléséhez felhasználtam egy erős közelmúltbeli földrengés beérkezett kérdőíveit, amelyeket a Kövesligethy Radó Szeizmológiai Obszervatórium bocsátott rendelkezésemre. A kiértékelés során tapasztalt nehézségeket felhasználtam arra, hogy tovább javítsam a kérdőívet, alkalmassá téve az automatikus kiértékelésre.

Végezetül statisztikai módszerekkel elemeztem az automatikus és a szakemberek által készített manuális kiértékelés különbségeit, és megvizsgáltam az eltérések valószínű okait.

Továbblépési lehetőségként két irányt azonosítottam be:

- a bizonytalan, illetve hiányos válaszokat a (Musson, 2006) által alkalmazott módszer helyett fuzzy logika használatával lehet kifinomultabban figyelembe venni,

az automatikus kiértékelést szükséges megtámogatni olyan módszerekkel, amelyek a kiugró értékeket jelzik, hogy a szakemberek dönthessenek arról, milyen korrekciót végezzenek, ennek érdekében ún. anomáliadetekciós („outlier detection”) algoritmusok segítségével kívánom vizsgálni a kérdőíves adatsorokat.

# A kiértékelőprogram vázlat

```
#####
```

```
### Modulok ###
```

```
#####
```

```
# a csv modul segít a CSV filok olvasásában és Unicode-ot is tud kezelni
```

```
import csv
```

```
#####
```

```
### Változók ###
```

```
#####
```

```
# Be és kimenő fájlok neve
```

```
file_be = "Croatia_20201229_1220_3050-6000_fontos_kerdesekek_unicode_2.txt"
```

```
# file_be = "Croatia_short.txt"
```

```
file_ki = "kiertekelt_eredmenyek_croatia.txt"
```

```
# ezekben a listákban lesznek a CSV fileből beolvasott kérdések rövidített nevei, és a  
megfigyelési eredmények
```

```
kerdeslista = []
```

```
eredmenylista = []
```

```
#####
```

```
### Filemuveletek ###
```

```
#####
```

```
# megnyitjuk írásra a kimenő file-t ahova a kiértékelési eredményeket írjuk majd
```

```
kiertekeles = open(file_ki, 'w', encoding="utf-8")
```

```
# megnyitjuk olvasásra a bemenő file-t (ez tab szeparált CSV file, jó Unicode kódolással kell  
megnyitni!)
```

```

with open(file_be, encoding="utf-16-le") as csvfile:

    f = csv.reader(csvfile, delimiter="\t")

    # elso sorban vannak a kerdesek rovid nevei

    kerdeslista = next(f)

    # a tovaabbi sorokban vannak az kerdoivek eredményei, amit beolvasunk egy listába
    eredmenyek=list(f)

# # Ellenorzés: milyen sorokat olvastunk be

# print("C-01 kerdeslista", kerdeslista)

# print("C-02 eredemenyek", eredmenyek) # exit()
#####

### Szöveges válaszok erosségekre konvertálása ###
#####

# segedváltozó, amivel észre fogjuk venni, ha megváltozik a település neve

elozo = "    # elozonek feldolgozott település neve

# végiglépkedünk a sorokon, és a mezokön: "a"-ban található az aktuálisan feldolgozott mezo a
feldolgozott sorban

for eredmeny in eredmenyek:

    for a in range(0, len(eredmeny)):

        # eltesszük a mezo eredetijét összehasonlításra

        eredeti = eredmeny[a]

        # 1,2,4 mezokat nem alakítjuk át (dátum, hányadik földrengés, irányítószám)

        if a == 1 or a == 2 or a == 4:

            continue

        # 3. mező a településnév, amit speciálisan kell kezelni

        if a == 3:

```

```

# gyakran van extra szóköz a településnév végén: a.strip() eltávolítja a whitespaceket
eredmeny[a]=eredmeny[a].strip()

# print(elozo, "-->", eredmeny[a])

# ha új települést találtunk beszúrunk egy "#" elválasztót az eredménylistába,
# így rögtön a legelső elé is bekerül egy felesleges "#", majd el kell távolítani
if eredmeny[a] != elozo:

    eredmenylista.append('#')

    elozo=eredmeny[a]

    # az érosségek szempontjából ezt a mezőt értelmetlen feldolgozni, ezért continue-
    val kiugrunk a belső ciklusból

    continue

# a többi kérdésre adott válaszokat számokká kódoljuk, amelyek már utalnak a rengés
erejére, de még nem közvetlenül EMS-98 intenzitások

# "nem tudja" típusú válaszok -> 0 (ezeknek nincs információ tartalma a rengés
megítélése szempontjából)

# "határozott nem" válaszok -> 1 (nem érezte, így valószínűleg nem volt akkora a rengés
intenzitása, ami a hatást kiváltotta volna)

# "határozott igen" válaszok -> 2, 3, 4, 5 (attól függően milyen érosségre utal a mezőben
eredetileg elhelyezkedő kifejezés)

# határozatlan válaszok -> 0 (nem tudja megítélni, illetve ébren volt, nem pedig a
rengésre ébredt fel, nincs többletinformáció a válaszból)

if eredmeny[a] == " " or "nem tudom" in eredmeny[a] or "ébren-awake" in eredmeny[a]:

    eredmeny[a] = '0'

# határozott nem válaszok -> 1

if "nem-no" in eredmeny[a] or "egy sem" in eredmeny[a] or "egyik sem" in eredmeny[a]
or "senki" in eredmeny[a] or 'nem volt' in eredmeny[a] or 'none' in eredmeny[a]:

```

eredmeny[a] = '1'

# erősség -> 2

if "igen" in eredmeny[a] and "néhány" not in eredmeny[a] or 'hullámzó mozgás' in eredmeny[a] or 'gyengén' in eredmeny[a] or 'alig' in eredmeny[a] or 'válaszfal' in eredmeny[a]:

eredmeny[a] = '2'

# erősség -> 3

if "néhány" in eredmeny[a] or 'kicsit' in eredmeny[a] or 'megrezdültek' in eredmeny[a] or 'zörrentek' in eredmeny[a] or 'hirtelen lökés' in eredmeny[a] or 'megmozdultak' in eredmeny[a] or 'gyenge' in eredmeny[a] or 'kimozdult' in eredmeny[a] or 'lehullott' in eredmeny[a] or 'nyílászáró' in eredmeny[a]:

eredmeny[a] = '3'

# erősség -> 4

if "sok" in eredmeny[a] or 'közepes' in eredmeny[a] or 'csörömpöltek' in eredmeny[a] or 'megrepedtek' in eredmeny[a] or 'rázkódás' in eredmeny[a] and 'nem volt' not in eredmeny[a] or 'elferdültek' in eredmeny[a] or 'elmozdultak' in eredmeny[a] or 'megsérültek' in eredmeny[a] or 'mennyezet' in eredmeny[a] or 'kémény' in eredmeny[a] or 'részben ledőlt' in eredmeny[a] or 'oromfalban' in eredmeny[a]:

eredmeny[a] = '4'

# erősség -> 5

if "mindenki" in eredmeny[a] or 'erős' in eredmeny[a] or 'erőteljesen' in eredmeny[a] or 'törték' in eredmeny[a] or 'legtöbb' in eredmeny[a] or 'felborultak' in eredmeny[a] or 'leestek' in eredmeny[a] or 'épületrész' in eredmeny[a] or 'lehullottak' in eredmeny[a] or 'ledőlt' in eredmeny[a] and 'részben' not in eredmeny[a] or 'most' in eredmeny[a] or 'főfal' in eredmeny[a]:

eredmeny[a] = '5'

# # Ellenorzés: melyik mezobol mi lett

```

#      print(a, eredeti, " => ", eredmény[a])

# a számokra kódolt sort beteszük az eredménylistába

eredmenylista.append(eredmeny)

# utolsó települést is megjelöljük # jellel, és a lista elejéről a felesleges #-t levesszük

eredmenylista.append('#')

eredmenylista.pop(0)

# Ellenorzés: egész átalakított eredménylistába mi került be?

# print("C-04 eredménylista",eredmenylista)

# exit()

#####

### Musson algoritmus ###

#####

# Segédváltozók

# Településenként haladunk,

telepules = " # az éppen feldolgozott település neve

N2 = 0      # a válaszadók teljes száma adott településről,

# 13 kategóriába rendezve gyűjtük az észlelt hatásokat, az Np és N1 listákban.

# A 13 kategória számozása a listákon belül:

# 0,1,2 - Épületkárok (0-enyhe, 1-közepes, 2-súlyos)

# 3,4,5 - Erős hatások (3-erős vibráció/hanghatás, 4-tárgyakat ért gyengébb hatások, pl.
eldőlnek/lengnek, 5- tárgyakat ért erősebb hatások, pl. bútorok dőlnek/tárgyak leesnek)

# 6,7,8 - Rémület (6-csak a megfigyelő, 7-néhányan mások is, 8-sokan mások)

# 9,10,11 - Gyenge hatások (8-érzékelhető mértéku rázkódás vagy zaj, 9-gyengén érezhető
rázkód's vagy zaj, 10-felébrednek emberek

# 12 - Maradék hatások (12-Enyhe zörgések, remegések)

```

# Np azok száma, akik határozott "igen"-t mondtak arra, hogy érzékelték a kategóriának megfelelő valamelyik hatást

Nplist = [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]

# N1 azok száma, akik határozott "igen"-t vagy "nem"-et mondtak (tehát nem voltak bizonytalanok)

N1list = [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]

# v 0 és 1 közötti paraméter. Megadja, mekkora arányban legyenek a "nem tudom" vagy bizonytalan kérdések "igen"-re, vagy "nem"-re konvertálva

v = 0.9

# Arra vagyunk kíváncsiak a 13 kategória mindegyikére, hogy az összes kitöltő mekkora százaléka érzékelt a kategóriába tartozó hatásokat

# az Rlist[] tárolja ezt a 13 arányszámot, amit az  $R = N_p / (N_1 + v(N_2 - N_1))$  képlet alapján számolunk ki

Rlist = [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]

upperfloor = 0 # azoknak a kitöltött kérdőíveknek a száma, amelyek a magasabb emeletekről (5. vagy magasabb) érkeztek, így a gyengébb rengést is jobban érezhették

sum1variable15 = 0 # ideiglenes változó a 15-ös szabályhoz: a földrengést senki sem érezte, vagy csak egy-két személy

sum2variable15 = 0 # ideiglenes változó a 15-ös szabályhoz: a földrengést csak néhány ember érezte, vagy csak emeleten tartózkodók

sum3variable15 = 0 # ideiglenes változó a 15-ös szabályhoz: a földrengést sokan érezték

# Településenként haladunk, hogy egy intenzitásértéket (Musson, 2006) szabályai szerint hozzárendeljünk a településről érkező bejelentésekhez

for e in eredmenylista:

    # új települést kezdünk és amíg nem érünk el egy '#' karakterhez, addig ugyanannak a településnek a bejelentéseit dolgozzuk fel

        if e != '#': # N2-ben számoljuk hány kérdőívet dolgozunk fel a településről

telepules = e[3]

N2 += 1

# Magasabb emeleten tartózkodott a bejelentő?

if '5th floor' in e[5] or '6th floor' in e[5]:

    # ha igen, azt megjegyezzük, mert ha csak magasabb emeletről jelentették,  
    az alacsonyabb intenzitást sejtet, mintha alacsonyabbról is

        upperfloor += 1

# Innentől az Np és N2 számlálókban számoljuk meg erősségek szerint a  
bejelenetett hatásokat, erőstől a gyenge fele haladva

# N1 akkor no, ha határozott igen/nem választ adott

# Np akkor no, ha érezte a hatást

# Épületkárok voltak-e?

# határozott "igen" választ adott az épületkár kérdésre, ezért N1 számlálót  
növeljük, de még nem tudjuk, mekkora lesz a kár nagysága az Np számlálókban

if e[27] == '2':

    N1list[0] += 1

    N1list[1] += 1

    N1list[2] += 1

    # súlyos épületkár -> 2-es számlálót növeljük

    if e[36] == '5' or e[37] == '4' or e[38] == '3' or e[38] == '4' or e[38] == '5'  
    or e[39] == '2' or e[39] == '3' or e[39] == '4' or e[39] == '5' or e[40] == '2':

        Nplist[2] += 1

    # közepes épületkár -> 1-es számlálót növeljük

    elif e[33] == '4' or e[33] == '5' or e[34] == '3' or e[34] == '4' or e[34] == '5'  
    or e[35] == '5' or e[36] == '4' or e[37] == '3' or e[38] == '2':

```

        Nplist[1] += 1

# enyhe épületkár -> 0-s számlálót növeljük

elif e[32] == '2' or e[32] == '3' or e[32] == '4' or e[32] == '5' or e[33] == '2'
or e[33] == '3' or e[34] == '2' or e[35] == '4' or e[36] == '3':

        Nplist[0] += 1

# határozott "nem" választ adott az épületkár kérdésére -> csak N1
számlálót növeljük, most nem kell kár nagyságával foglalkozni

elif e[27] == '1':

        N1list[0] += 1

        N1list[1] += 1

        N1list[2] += 1

# Rázkódások voltak-e?

# Határozott "igen" választ adott

if e[7] != '0' and e[7] != '1':

        Nplist[9] += 1

        N1list[9] += 1

if e[8] == '4' or e[8] == '5': # a földrengést erősnek írták le

        Nplist[3] += 1

        N1list[3] += 1

        N1list[10] += 1

elif e[8] == '2' or e[8] == '3': # a földrengést gyengének írták le

        N1list[3] += 1

        Nplist[10] += 1

        N1list[10] += 1

elif e[7] == '1' or e[7] == '0': # nem volt rázkódás, vagy bizonytalanok ezzel kapcsolatban

```

N1list[3] += 1

N1list[9] += 1

N1list[10] += 1

# Tárgyak elmozdultak-e?

# Elobb azokat nézzük meg amikor igen, tárgyak elmozdultak:

if e[15] != '0' or e[15] != '1' and e[16] != '0' or e[16] != '1' and e[17] != '0' or e[17] != '1'  
and e[18] != '0' or e[18] != '1' and e[19] != '0' or e[19] != '1' and e[20] != '0' or e[20] != '1'  
and e[21] != '0' or e[21] != '1' and e[22] != '0' or e[22] != '1' and e[23] != '0' or e[23] != '1'  
and e[24] != '0' or e[24] != '1' and e[25] != '0' or e[25] != '1' and e[26] != '0' or e[26] !=  
'1':

#igen, beszámoltak tárgyakat ért hatásokról, megnézzük, mekkora volt ez a hatás

if e[15] == '5' or e[16] == '5' or e[17] == '4' or e[17] == '5' or e[18] == '5' or e[19]  
== '4' or e[20] == '5' or e[21] == '4' or e[21] == '5' or e[22] == '2' or e[23] == '2' or e[24]  
== '2' or e[25] == '2' or e[26] == '5' and e[15] != '0' or e[15] != '1' or e[15] != '2' or e[15]  
!= '3' and e[16] != '0' or e[16] != '1' or e[16] != '3' and e[17] != '0' or e[17] != '1' or e[17]  
!= '3' and e[18] != '0' or e[18] != '1' or e[18] != '3' and e[19] != '0' or e[19] != '1' or e[19]  
!= '3':

# a tárgyakat ért hatások erősek voltak, a megfelelő számlálót növeljük

Nplist[5] += 1

N1list[5] += 1

N1list[4] += 1

elif e[15] == '2' or e[15] == '3' or e[15] == '4' and e[16] == '3' or e[16] == '4' and  
e[17] == '3' or e[18] == '3' or e[18] == '4' and e[19] == '3' or e[20] == '3' or e[20] == '4' or  
e[21] == '3' or e[26] == '4':

#a tárgyakat ért hatások gyengék voltak

Nplist[4] += 1

N1list[4] += 1

N1list[5] += 1

# Most azokat nézzük meg, amikor nem mozdultak el a tárgyak

if e[15] == '0' or e[15] == '1' and e[16] == '0' or e[16] == '1' and e[17] == '0' or  
e[17] == '1' and e[18] == '0' or e[18] == '1' and e[19] == '0' or e[19] == '1' and e[20] ==  
'0' or e[20] == '1' and e[21] == '0' or e[21] == '1' and e[22] == '0' or e[22] == '1' and e[23]  
== '0' or e[23] == '1' and e[24] == '0' or e[24] == '1' and e[25] == '0' or e[25] == '1' and  
e[26] == '0' or e[26] == '1':

# arról számoltak be, hogy egyetlen tárgynál sem tapasztaltak semmilyen  
hatást, vagy bizonytalanok ezzel kapcsolatban

N1list[4] += 1

N1list[5] += 1

# Élőlényeket érte-e valami hatás, megijedtek-e?

if e[9] != '0' or e[12] != '0' or e[13] != '0' or e[14] != '0':

# igen, beszámoltak a környező személyeket vagy állatokat ért hatásokról,  
megnézzük, mekkora hatásról:

if e[9] == '4' or e[9] == '5' or e[11] == '4' or e[11] == '5' or e[12] == '5' or  
e[13] == '4':

# a rengéstől sokan megijedtek a megfigyelő környezetében

Nplist[8] += 1

N1list[6] += 1

N1list[7] += 1

N1list[8] += 1

elif e[9] == '3' or e[11] == '3' or e[12] == '3' or e[13] == '3' or e[14] == '2'  
and e[11] != '4' and e[11] != '5' and e[12] != '5' and e[13] != '4':

#néhányan megijedtek a közelben#

Nplist[7] += 1

```

        N1list[6] += 1

        N1list[7] += 1

        N1list[8] += 1

elif e[0] == '2' or e[9] == '0' or e[9] == '1' or e[10] == '2' or e[11] == '0' or
e[11] == '1' or e[12] == '1' or e[12] == '2' or e[13] == '1':

    # a megfigyelő megijedt, de más ember megijedéséről a beszámoló
    nem szól

    Nplis[6] += 1

    N1list[6] += 1

    N1list[7] += 1

    N1list[8] += 1

elif e[0] == '1' or e[9] == '1' and e[10] == '1' or e[11] == '1' or e[12] == '1'
or e[13] == '1' or e[14] == '1':

    # senki sem érezte a földrengést a válaszadó környezetében

    N1list[6] += 1

    N1list[7] += 1

    N1list[8] += 1

# Valaki felébredt-e?

if e[10] == '2' and e[11] != '0' and e[11] != '1':

    #Igen, felébredt a kitöltő, vagy a környezetében alvó emberek

    Nplis[11] += 1

    N1list[11] += 1

elif e[10] == '0' or e[10] == '1' and e[11] == '0' or e[11] == '1':

    # Nem, senki sem ébredt fel a földrengésre

    N1list[11] += 1

```

```

# tárgyak zörögtek-e, rázkódtak-e?

if e[16] == '3' or e[16] == '4' or e[17] == '3' or e[22] == '2':

    #igen, megfigyelték a tárgyak zörgését vagy rázkódását

    Nplist[12] += 1

    N1list[12] += 1

elif e[16] == '0' or e[16] == '1' and e[17] == '0' or e[17] == '1' and e[22] == '0' or
e[22] == '1':

    #Nem, nem tapasztalták a tárgyak zörgését/rázkódását

    N1list[12] += 1

# Hányan érezték a rengést?

if e[9] == '0' or e[9] == '1':

    #a rengést csak a megfigyelő érezte, esetleg egy-ketten rajta kívül

    sum1variable15 += 1

elif e[9] == '3' or 'emeleten' in e[5]:

    #a rengést néhányan érezték, vagy csak az emeleten tartózkodók#

    sum2variable15 += 1

elif e[9] == '4' or e[9] == '5' or 'szabadban' not in e[5] or "járműben" not in e[5]:

    #a rengést sokan, a legtöbben, vagy akár mindenki érezte, illetve csak
    beltéren tartózkodók érezték#

    sum3variable15 += 1

# Ha elértünk egy '#' karaktert, akkor végére értünk a településnek, kiszámolhatjuk rá az
arányokat

if e == '#':

    # végiglépünk a 13 kategórián:

    for f in range(0, len(Rlist)):

```

```
Rlist[f] = Nplist[f]/(N1list[f] + v*(N2 - N1list[f]))
```

```
#megfigyelési arányszámok kiszámítása minden kategória megtörtént  
print(Rlist)
```

```
# a szabályokhoz áttérünk a B, S, F, O, R kategórianévek használatára, ami  
(Musson, 2006) cikkében szerepel:
```

```
B1 = Rlist[0] #EMS-skála szerinti 1-es fokozatú épületkár
```

```
B2 = Rlist[1] #EMS-skála szerinti 2-es fokozatú épületkár
```

```
B3 = Rlist[2] #EMS-skála szerinti 3-as fokozatú épületkár
```

```
S1 = Rlist[3] #a földrengés okozta vibráció erősnek vagy a hallott zaj hangosnak  
van leírva
```

```
S2 = Rlist[4] #kisebb mértékű erős hatások, például képek lengedezése vagy  
kisebb tárgyak leesése
```

```
S3 = Rlist[5] #nagyobb mértékű erős hatások, például bútorok felborulása vagy  
könyvek leesése a polcokról
```

```
F1 = Rlist[6] #a megfigyelő megijedt
```

```
F2 = Rlist[7] #néhányan megijedtek a közelben
```

```
F3 = Rlist[8] #sokan megijedtek a közelben
```

```
O1 = Rlist[9] #a rázkódást vagy zajt taglaló kérdésekre néhány "igen" válasz
```

```
O2 = Rlist[10] #a rázkódást gyengének vagy a zajt enyhének írják le
```

```
O3 = Rlist[11] #az alvó emberek felébredtek
```

```
R1 = Rlist[12] #a tárgyak zörögtek vagy rázkódtak
```

```
if Nplist[0] >= 4 and B2 >= 0.6 and Nplist[1] != 0 and N2 != upperfloor:
```

```
#Musson 1. szabálya#
```

```
print(f'Az intenzitás értéke {telepules} településen 8.')
```

```
kiertekeles.write(f'{telepules}\t8\n')
```

```
elif Nplist[1] >= 4 and B3 >= 0.2 and N2 != upperfloor: #Musson 2. szabálya#
```

```
    print(f'Az intenzitás értéke {telepules} településen 8.')
```

```
    kiertekeles.write(f'{telepules}\t8\n')
```

```
elif Nplist[0] >= 4 and B2 >= 0.2 and B1 >= 0.6 and N2 != upperfloor: #Musson  
3. szabálya#
```

```
    print(f'Az intenzitás értéke {telepules} településen 7.')
```

```
    kiertekeles.write(f'{telepules}\t7\n')
```

```
else:
```

```
    P6 = 0 #6-os intenzitás kiértékelésének segédváltozója
```

```
    P5 = 0 #5-ös intenzitás kiértékelésének segédváltozója
```

```
    if B2 > 0 or B3 > 0:
```

```
        P6 += 1
```

```
    if B1 > 0.2:
```

```
        P6 += 2
```

```
    if 0 < B1 <= 0.2 and B2 == 0 and B3 == 0:
```

```
        P5 += 1
```

```
    #felső három ciklus Musson 4. szabályából
```

```
    if B1 == 0 and B2 == 0 and B3 == 0:
```

```
        P6 -= 2 #Musson 5. szabálya#
```

```
    if S1 > 0.4:
```

```
        P5 += 1
```

```
    if S1 > 0.8:
```

```
        P6 += 1
```

```
    if S1 == 0:
```

P5 -= 1

P6 -= 2

#felső három ciklus Musson 6. szabályából#

if S2 > 0.2: P5 += 1 if S2 > 0.6:

P6 += 2

if S2 == 0:

P5 -= 1

P6 -= 2

#felső három ciklus Musson 7. szabályából#

if S3 > 0:

P5 += 1

if S3 > 0.2 and Nplist[5] > 1:

P6 += 2

if S3 == 0:

P6 -= 1

#felső három ciklus Musson 8. szabályából#

if F1 > 0.2:

P5 += 1

if F1 > 0.6:

P6 += 2

if F1 == 0:

P5 -= 1

P6 -= 2

#felső három ciklus Musson 9. szabályából#

```

if F2 > F3:
    P5 += 2

if F2 < F3:
    P6 += 2

#felső két ciklus Musson 10. szabályából#

if F3 > 0.2:
    P6 += 1 #Musson 11. szabálya#

if F2 == 0:
    P5 -= 1
    P6 -= 2

#felső két ciklus Musson 12. szabályából#

if F3 == 0:
    P6 -= 1

if P6 > P5 and P6 >= 4 and N2 != upperfloor: #Musson 13. szabálya#
    print(f'Az intenzitás értéke {telepules} településen 6.')
    kiertekeles.write(f'{telepules}\t6\n')
    print(P5, P6)

elif P6 <= P5 and 0 < P6 < 4 and N2 != upperfloor:
    P5 += 1

#ez és a következő ciklus Musson 14. szabályából#

if P5 >= 3:
    print(f'Az intenzitás értéke {telepules} településen 5.')
    kiertekeles.write(f'{telepules}\t5\n')
    print(P5, P6)

```

if  $P6 \leq P5$  and  $P6 \geq 4$  or  $P6 > P5$  and  $0 < P6 < 4$  or  $P6 \leq 0$  or  $P5 < 3$   
and  $0 < P6 < 4$  or  $N2 == \text{upperfloor}$ :

print(P5, P6)

P2 = sum1variable15

P3 = sum2variable15

P4 = sum3variable15

#a felső három segédváltozó a 2-es, 3-as, 4-es intenzitások  
meghatározásához attól függően választódik ki, hányan érezték a rengést,  
ez Musson 15. szabálya is egyben#

if  $P3 < P2$  and  $P4 < P2$ :

P2 = 2

P3 = 0

P4 = 0

elif  $P3 < P2 == P4$  or  $P4 < P2 == P3$ :

P2 = 1

P3 = 2

P4 = 0

elif  $P3 > P4$ :

P2 = 0

P3 = 2

P4 = 0

else:

P2 = 0

P3 = 0

P4 = 4

#felső négy ciklus Musson 16. szabályából#

print(P2, P3, P4)

if P5 > 0: #Musson 17. szabálya#

P4 += P5

if O1 <= 0.1:

P2 += 1

if 0.1 < O1 <= 0.4:

P3 += 1

P2 -= 1

if O1 > 0.4:

P4 += 1

P3 -= 1

P2 -= 2

#felső három ciklus Musson 18. szabályából#

if O2 > 0.8:

P4 -= 1

if P2 > P3:

P2 += 1

if P3 >= P2:

P3 += 1

#felső ciklusok Musson 19. szabályából#

if O3 > 0.2: #Musson 20. szabálya#

P4 += 1

if R1 >= 0.2:

```

        P4 += 1

if R1 > 0.4:

    P4 += 1

    P2 -= 1

#felső két ciklus Musson 21. szabályából#

if R1 == 0: #Musson 22. szabálya#

    P4 -= 1

if N2 == upperfloor: #Musson 23. szabálya#

    P2 += 2

if P2 > P3 and P2 > P4:

    print(f'Az intenzitás értéke {telepules} településen 2.')

    kiertekeles.write(f'{telepules}\t2\n')

#ez és a következő ciklus Musson 24. szabályából#

elif P4 > P2 and P4 > P3 and N2 != upperfloor:

    print(f'Az intenzitás értéke {telepules} településen 4.')

    kiertekeles.write(f'{telepules}\t4\n')

else: #Musson 25. szabálya#

    print(f'Az intenzitás értéke {telepules} településen 3.')

    kiertekeles.write(f'{telepules}\t3\n')

#intenzitások segédváltozóinak kiírása a program ellenőrzéséhez:

print("[P2, P3, P4, P5, P6] -> [", P2, P3, P4, P5, P6, "]\n")

# lenullázzuk a segédváltozók értékeit minden egyes újabb település előtt
Rlist = [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]

Nplist = [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]

```

```
N1list = [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
```

```
N2 = 0
```

```
upperfloor = 0
```

```
sum1variable15 = 0
```

```
sum2variable15 = 0
```

```
sum3variable15 = 0
```

```
# lezárjuk az írásra megnyitott file-t
```

```
kiertekeles.close()
```