



## PŘÍLOHA 1

"""

Vytvořeno 30.12.20 16:53

@autor: Martin Lebeda

Výpočet tenzoru deformace podle článku Oda a kol. 1993.

Vstup: DFN model ve formátu vtk.

Výstup: Tensor deformace vypsáný do konzole.

"""

```
import numpy as np
import vtk
# funkce pro vytvoření puklinových tenzorů
from DFN_tools_fcns_update import crack_tensors

path = "C:\\\\" # absolutní cesta k vtk souboru
invtkfilename = path+"Filename.vtk" # vtk soubor s DFN modelem

### Načtení souboru vtk
reader = vtk.vtkPolyDataReader()
reader.SetFileName(invtkfilename)
reader.ReadAllScalarsOn()
reader.Update()

dfn_data = reader.GetOutput() # všechny data

### Počet puklin
frcs_number = dfn_data.GetNumberOfCells()
print(frcs_number)
```



```
### Výpočet
kn = 1.35615 # [-], parametr normálové tuhosti
ks = 1.188516 # [-], parametr smykové tuhosti
P_num = 0.164 # [-], Poissonovo číslo
Young_mod = 44000 # [MPa], modul pružnosti matrice
sigma = np.array([[1,0,0], [0,0,0], [0,0,0]]) # [MPa], tenzor napětí
K_delta = np.array([[1,0,0], [0,1,0], [0,0,1]]) # Kroneckerovo delta

Volume = 1.4**3 # [m3], objem modelované horniny
F_temp = crack_tensors(fracs_number, dfn_data)
F_2 = F_temp[0] / Volume # puklinový tenzor 2. řádu
F_4 = F_temp[1] / Volume # puklinový tenzor 4. řádu

part1_temp = (1 + P_num) * np.einsum('ik,jl->ikjl', K_delta,
K_delta)
part2_temp = P_num * np.einsum('ij,kl->ijkl', K_delta, K_delta)
part3_temp = (1/kn - 1/ks) * F_4
part4_temp = (1 / (4 * ks)) * np.einsum('ik,jl->ikjl', K_delta, F_2)
part5_temp = (1 / (4 * ks)) * np.einsum('jk,il->jkil', K_delta, F_2)
part6_temp = (1 / (4 * ks)) * np.einsum('il,jk->iljk', K_delta, F_2)
part7_temp = (1 / (4 * ks)) * np.einsum('jl,ik->jlik', K_delta, F_2)

part1 = np.einsum('ikjl,kl->ij', part1_temp, sigma)
part2 = np.einsum('ijkl,kl->ij', part2_temp, sigma)
part3 = np.einsum('ijkl,kl->ij', part3_temp, sigma)
part4 = np.einsum('ikjl,kl->ij', part4_temp, sigma)
part5 = np.einsum('jkil,kl->ij', part5_temp, sigma)
part6 = np.einsum('iljk,kl->ij', part6_temp, sigma)
part7 = np.einsum('jlik,kl->ij', part7_temp, sigma)

# tenzor deformace
eps_ij = (1 / Young_mod) * (part1 - part2 + part3 + part4 + part5 +
part6 + part7)
print(eps_ij)
```