

Trout Documentation

1. `skincav(cavgroup,cavslot,mediumgroup,mediumslot,newgroup,newslot)`

1.1. Version: 1.0

1.2. Date: 2022/08/31

1.3. Description:

Create a new group for the zones on the skin of a group. Good for MIT skin plots.

1.4. Inputs:

cavgroup: cavern group

cavslot: cavern slot

mediumgroup: medium group that cavgroup zones are also grouped into e.g. ('Salt')

mediumslot: medium slot that the cavgroup zones are also grouped into

newgroup: new group name

newslot: new slot name

1.5. Returns:

N/A

1.6. Example:

Console:

```
flac3d> [skincav('Cav1','2008',' Salt','Materials','Cav1Skin','Skins')]
```

Cavern Skin Shown in Contour Plot for Least Compressive Principle Stress



2. `lodeang(s1,s2,s3)`

2.1. Version: 1.0

2.2. Date: 2022/08/31

2.3. Description:

Calculate the lode angle given principal stresses (sign convention: (-) compression)

2.4. Inputs:

s1: maximum principal stress

s2: intermediate principal stress

s3: minimum principal stress

2.5. Returns:

Lode angle (radians)

2.6. Example:

Console:

```
flac3d> [lodeang(-1.0e6,-2.0e6,-20.0e6)]  
0.476821
```

3. **rdc(s1,s2,s3,D1,D2,n,T0,s0)**

3.1. Version: 1.0

3.2. Date: 2022/08/31

3.3. Description:

3.4. Calculate the factor of safety against the RESPEC dilation criterion (sign convention: (-) compression)

3.5. Inputs:

s1: maximum principal stress

s2: intermediate principal stress

s3: minimum principal stress

D1: D1

D2: D2

n: n

T0: T0

3.6. Returns:

Factor of safety against RESPEC dilation criterion

3.7. Example:

Code:

```
fish define CalcExtras()  
  local m, D1, D2, T0, n, s0, PS1, PS2, PS3  
  m=1/144  
  loop foreach local zp zone.list()  
    PS1 = zone.stress.prin.x(zp)*m  
    PS2 = zone.stress.prin.y(zp)*m  
    PS3 = zone.stress.prin.z(zp)*m  
    if zone.group(zp,'Mats')=='Salt'  
      zone.extra(zp,1)=rdc(PS1,PS2,PS3,0.779,1.174,0.839,227,1)  
    endif  
  endloop  
end  
@CalcExtras()
```

4. **mcfs3d(s1,s2,s3,Co,phi)**

4.1. Version: 1.0

4.2. Date: 2022/08/31

4.3. Description:

4.4. Calculate the factor of safety against the 3D mohr-coulomb failure criterion (sign convention: (-) compression)

4.5. Inputs:

s1: maximum principal stress

s2: intermediate principal stress

s3: minimum principal stress

Co: Cohesion

phi: friction angle (radians)

4.6. Returns:

Factor of safety against the 3D mohr-coulomb failure criterion

4.7. Example:

Code:

```
fish define CalcExtras()  
  local m  
  m=1/144  
  loop foreach local zp zone.list()  
    PS1 = zone.stress.prin.x(zp)*m  
    PS2 = zone.stress.prin.y(zp)*m  
    PS3 = zone.stress.prin.z(zp)*m  
    if zone.group(zp,'Mats')== 'Dolomite'  
      zone.extra(zp,1)=mcfs3d(PS1,PS2,PS3,1000,30*math.pi/180)  
    endif  
  endloop  
end  
@CalcExtras()
```

5. **dp(s1,s2,s3,lim)**

5.1. Version: 1.0

5.2. Date: 2022/08/31

5.3. Description:

5.4. Calculate the factor of safety against dilation potential (sign convention: (-) compression)

5.5. Inputs:

s1: maximum principal stress

s2: intermediate principal stress

s3: minimum principal stress

lim: dilation limit (maximum $\frac{\sqrt{J_2}}{I_1}$ value) e.g.(0.18,0.27,0.54)

5.6. Returns:

Factor of safety against dilation potential

5.7. Example:

Code:

```
fish define CalcExtras()  
  local m, D1, D2, T0, n, s0, PS1, PS2, PS3  
  m=1/144  
  loop foreach local zp zone.list()  
    PS1 = zone.stress.prin.x(zp)*m  
    PS2 = zone.stress.prin.y(zp)*m
```

```
    PS3 = zone.stress.prin.z(zp)*m
    if zone.group(zp,'Mats')== 'Salt'
        zone.extra(zp,1)=dp(PS1,PS2,PS3,0.18)
        zone.extra(zp,2)=dp(PS1,PS2,PS3,0.27)
    endif
endloop
end
@CalcExtras()
```