



**Class 1: (Planar finite slides)**

**Known variables**

H, beta (\*), DB(\*),BC(\*)

**Unknown variables**

Theta, L

**Case A: Known beta and known BC**

1. Compute DB
2. Compute DC
3. Compute L
4. BC may be directly measured or iteratively computed
5. Compute area of the sliding mass
6. Compute weight of the sliding mass
7. Compute driving and resisting forces
8. Compute factor of safety

**Case B: Unknown beta and known C**

**Default**: postulate FS = 1, compute backwards beta

Set beta (design) < beta (computed) NB: Default corresponds to limiting equilibrium conditions

**Customize** : FS = User defined value

**Case C: Known beta and unknown C**

Postulate a value for BC, Compute FS,

If FS > 1+0.001;

Update BC, BC = BC -delta BC, and recompute FS

Iterate until FS = 1+/- 0.001;

Else if FS < 1-0.001;

Update BC, BC = BC +delta BC, and recompute FS

Iterate until FS = 1+/- 0.001;

Else

Print critical location of tension crack has been found.

**NB**: this version does don’t incorporate the depth of tension crack, think about how we can appropriately add that to the geometry set up

Also think about how we appropriately define delta BC, to ensure computational efficiency and avoid getting stuck in an infinite loop.

This is my thinking, I would be very happy if you guys disagree and offer better ways of approaching the code. Thanks