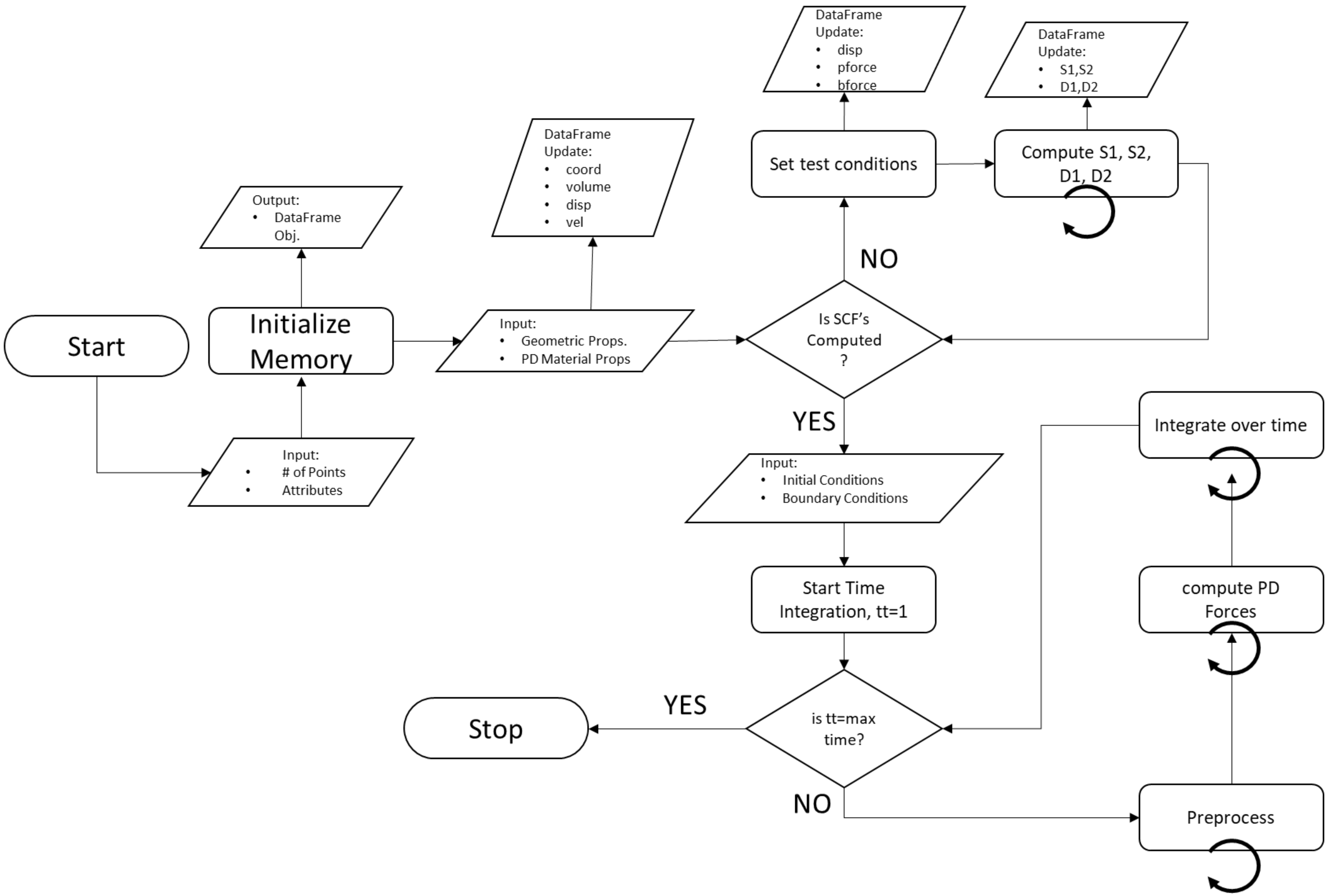
Start
Input:
# of Points
Attributes
Initialize Memory
Output:
DataFrame Obj.
Input:
Geometric Props.
PD Material Props
DataFrame Update:
coord
volume
disp
vel
Is SCF's Computed ?
NO
Set test conditions
DataFrame Update:
disp
pforce
bforce
Compute S1, S2, D1, D2
DataFrame Update:
S1,S2
D1,D2
YES
Input:
Initial Conditions
Boundary Conditions
Start Time Integration, tt=1
is tt=max time?
YES
Stop
NO
Preprocess
compute PD Forces
Integrate over time

Compute Surface Correction Factors:			
Require:		δ	Horizon Parameter
Require:		a, b, d	PD Material Properties
Require:		μ	Shear Modulus
Require:	Pandas DataFrame Object		Instance that implements required data structure for PD attributes for all points
1.	**for each** material point i in the body **do**		
2.	Get neighbor indices of i^{th} material point		$\leftarrow j, \quad if: \lvert \boldsymbol{x_j} - \boldsymbol{x_i} \rvert \le \delta$
3.	Compute initial distance, $\lvert \boldsymbol{\xi} \rvert$		$\leftarrow \lvert \xi_{ij} \rvert = \lvert \boldsymbol{x_j} - \boldsymbol{x_i} \rvert$
4.	Compute length of the deformed bond, $\lvert \boldsymbol{\eta} \rvert$		$\leftarrow \lvert \eta_{ij} \rvert = \lvert (\boldsymbol{x_j} + \boldsymbol{u_j}) - (\boldsymbol{x_i} + \boldsymbol{u_i}) \rvert$
5.	Compute stretch of the deformed bond, $\lvert \boldsymbol{s} \rvert$		$\leftarrow s_{ij} = \frac{\lvert \boldsymbol{\eta_{ij}} \rvert}{\lvert \boldsymbol{\xi_{ij}} \rvert} = \frac{\lvert (\boldsymbol{x_j} + \boldsymbol{u_j}) - (\boldsymbol{x_i} + \boldsymbol{u_i}) \rvert}{\lvert \boldsymbol{x_j} - \boldsymbol{x_i} \rvert}$
6.	Compute Volume Correction Factor, $v_{c(j)}$		$\leftarrow v_c = \begin{cases} \frac{(\delta + r - \lvert \boldsymbol{\xi_{ij}} \rvert)}{\Delta}, & \delta - \frac{\Delta}{2} < \lvert \boldsymbol{\xi_{ij}} \rvert < \delta, \\ 1, & \lvert \boldsymbol{\xi_{ij}} \rvert \le \delta - \frac{\Delta}{2} \\ 0, & \lvert \boldsymbol{\xi_{ij}} \rvert > \delta \end{cases}$
7.	Compute parameter Λ		$\leftarrow \Lambda_{\text{ij}} = \left(\frac{\boldsymbol{\xi_{ij}}}{\lvert \boldsymbol{\xi_{ij}} \rvert} \right) \cdot \left(\frac{\boldsymbol{\eta_{ij}}}{\lvert \boldsymbol{\eta_{ij}} \rvert} \right)$
8.	Compute Dilatation Term, $\boldsymbol{\theta}$		$\leftarrow \theta_i = d\delta \sum_{j=1}^{N} s_{ij} \Lambda_{ij} v_{c(j)} V_j$
9.	Compute Strain Energy Density, W		$\leftarrow W_i = a\theta_i^2 + b\delta \sum_{j=1}^{N} \frac{(\lvert \boldsymbol{\eta_{ij}} \rvert - \lvert \boldsymbol{\xi_{ij}} \rvert)^2}{\lvert \boldsymbol{\xi_{ij}} \rvert} v_{c(j)} V_j$
10.	Set Surface Correction Factor, S_{i_α} or D_{i_α}		$\leftarrow S_{i_\alpha}$ or $D_{i_\alpha} = \frac{W^{CM}}{W_i^{PD}}$ or $\frac{\theta^{CM}}{\theta_i^{PD}}$

Preprocess with Surface Correction Vectors:			
Require:		δ	Horizon Parameter
Require:		a, b, d	PD Material Properties
Require:	Pandas DataFrame Object		Instance that implements required data structure for PD attributes for all points
1.	**for each** material point i in the body **do**		
2.	Get neighbor indices of i^{th} material point		$\leftarrow j, \quad if: \ \lvert \boldsymbol{x_j} - \boldsymbol{x_i} \rvert \le \delta$
3.	Compute Principal Axes for Surface Correction Ellipsoid, G_b, G_d		$\leftarrow G_{b_{ij}} \& G_{d_{ij}}$
4.	Compute initial distance, $\lvert \boldsymbol{\xi} \rvert$		$\leftarrow \lvert \xi_{ij} \rvert = \lvert \boldsymbol{x_j} - \boldsymbol{x_i} \rvert$
5.	Compute length of the deformed bond, $\lvert \boldsymbol{\eta} \rvert$		$\leftarrow \lvert \eta_{ij} \rvert = \lvert (\boldsymbol{x_j} + \boldsymbol{u_j}) - (\boldsymbol{x_i} + \boldsymbol{u_i}) \rvert$
6.	Compute stretch of the deformed bond, $\lvert \boldsymbol{s} \rvert$		$\leftarrow s_{ij} = \dfrac{\lvert \boldsymbol{\eta_{ij}} \rvert}{\lvert \boldsymbol{\xi_{ij}} \rvert} = \dfrac{\lvert (\boldsymbol{x_j} + \boldsymbol{u_j}) - (\boldsymbol{x_i} + \boldsymbol{u_i}) \rvert}{\lvert \boldsymbol{x_j} - \boldsymbol{x_i} \rvert}$
7.	Compute Volume Correction Factor, $v_{c(j)}$		$\leftarrow v_c = \begin{cases} \dfrac{(\delta + r - \xi_{ij})}{\Delta}, & \delta - \dfrac{\Delta}{2} < \xi_{ij} < \delta, \\ 1, & \xi_{ij} \le \delta - \dfrac{\Delta}{2} \\ 0, & \xi_{ij} > \delta \end{cases}$
8.	Compute parameter Λ		$\leftarrow \Lambda_{\mathrm{ij}} = \left(\dfrac{\boldsymbol{\xi_{ij}}}{\lvert \boldsymbol{\xi_{ij}} \rvert} \right) \cdot \left(\dfrac{\boldsymbol{\eta_{ij}}}{\lvert \boldsymbol{\eta_{ij}} \rvert} \right)$
9.	Compute Dilatation Term, $\boldsymbol{\theta}$		$\leftarrow \theta_i = d\delta \sum_{j=1}^{N} G_{d_{ij}} s_{ij} \Lambda_{ij} v_{c(j)} V_j$
10.	Compute Strain Energy Density, W		$\leftarrow W_i = a\theta_i^2 + b\delta \sum_{j=1}^{N} G_{b_{ij}} \dfrac{\left(\lvert \boldsymbol{\eta_{ij}} \rvert - \lvert \boldsymbol{\xi_{ij}} \rvert \right)^2}{\lvert \boldsymbol{\xi_{ij}} \rvert} v_{c(j)} V_j$

Compute PD Forces and Iteration with Adaptive Dynamic Relaxation:		
Require:	δ	Horizon Parameter
Require:	a, b, d	PD Material Properties
Require:	Pandas DataFrame Object	Instance that implements required data structure for PD attributes for all points
1.	**for each** iteration between 1 to max. iter. **do**	
2.	**for each** material point i in the body **do**	
3.	Get neighbor indices of i^{th} material point	$\leftarrow j, \quad if: \lvert x_j - x_i \rvert \leq \delta$
4.	Compute Principal Axes for Surface Correction Ellipsoid, G_b, G_d	$\leftarrow G_{b_{ij}} \& G_{d_{ij}}$
5.	Compute initial distance, $\lvert\boldsymbol{\xi}\rvert$	$\leftarrow \lvert\xi_{ij}\rvert = \lvert\boldsymbol{x_j} - \boldsymbol{x_i}\rvert$
6.	Compute length of the deformed bond, $\lvert\boldsymbol{\eta}\rvert$	$\leftarrow \lvert\eta_{ij}\rvert = \lvert(\boldsymbol{x_j} + \boldsymbol{u_j}) - (\boldsymbol{x_i} + \boldsymbol{u_i})\rvert$
7.	Compute stretch of the deformed bond, $\lvert\boldsymbol{s}\rvert$	$\leftarrow s_{ij} = \frac{\lvert\boldsymbol{\eta_{ij}}\rvert}{\lvert\boldsymbol{\xi_{ij}}\rvert} = \frac{\lvert(\boldsymbol{x_j} + \boldsymbol{u_j}) - (\boldsymbol{x_i} + \boldsymbol{u_i})\rvert}{\lvert\boldsymbol{x_j} - \boldsymbol{x_i}\rvert}$
8.	Compute Volume Correction Factor, $v_{c(j)}$	$\leftarrow v_c = \begin{cases} \frac{(\delta + r - \xi_{ij})}{\Delta}, & \delta - \frac{\Delta}{2} < \xi_{ij} < \delta, \\ 1, & \xi_{ij} \leq \delta - \frac{\Delta}{2} \\ 0, & \xi_{ij} > \delta \end{cases}$
9.	Compute parameter Λ	$\leftarrow \Lambda_{\mathrm{ij}} = \left(\frac{\boldsymbol{\xi_{ij}}}{\lvert\boldsymbol{\xi_{ij}}\rvert}\right) \cdot \left(\frac{\boldsymbol{\eta_{ij}}}{\lvert\boldsymbol{\eta_{ij}}\rvert}\right)$
10.	Compute PD Force $\boldsymbol{t_{ij}}$	$\leftarrow \boldsymbol{t_{ij}} = 2\delta\left(G_{d_{ij}} d \frac{\Lambda_{ij}}{\lvert\boldsymbol{\xi_{ij}}\rvert}(a\theta_i) + G_{b_{ij}} b(s_{ij})\right)\frac{\boldsymbol{\eta_{ij}}}{\lvert\boldsymbol{\eta_{ij}}\rvert}$
11.	Compute PD Force $\boldsymbol{t_{ji}}$	$\leftarrow \boldsymbol{t_{ji}} = -2\delta\left(G_{d_{ij}} d \frac{\Lambda_{ij}}{\lvert\boldsymbol{\xi_{ij}}\rvert}(a\theta_j) + G_{b_{ij}} b(s_{ij})\right)\frac{\boldsymbol{\eta_{ij}}}{\lvert\boldsymbol{\eta_{ij}}\rvert}$
12.	Summation of PD Forces	$\leftarrow \sum_{j=1}^{N}(\boldsymbol{t_{ij}} - \boldsymbol{t_{ji}}) v_{c(j)} V_j$
13	Compute Elements of Stiffness Matrix, $\sum_{j=1}^{N}\lvert K_{ij}\rvert$	$\leftarrow \sum_{j=1}^{N}\lvert K_{ij}\rvert = \sum_{j=1}^{N}\frac{\lvert\boldsymbol{\xi_{ij}} \cdot \boldsymbol{e}\rvert}{\lvert\boldsymbol{\xi_{ij}}\rvert}\frac{4\delta}{\lvert\boldsymbol{\xi_{ij}}\rvert}\left(\frac{1}{2}\frac{ad^2\delta}{\lvert\boldsymbol{\xi_{ij}}\rvert}\left(v_{c(i)}V_i + v_{c(j)}V_j\right) + b\right)$

14.	Compute Diagonal Element of Density Matrix, λ_{ii}	$\leftarrow \lambda_{ii} = \frac{1}{4}\Delta t^2 \sum_{j=1}^{N} \lvert K_{ij} \rvert$
15.	Initialize $c_n = 0, c_{num} = 0, c_{denom} = 0$	
16.	**for each** material point i in the body **do**	
17.	Compute Diagonal Element of Local Stiffness Matrix, ${}^1K_{ii}^n$	$\leftarrow {}^1K_{ii}^n = -\left(\frac{\frac{\left(\sum_{j=1}^{N}(\boldsymbol{t_{ij}} - \boldsymbol{t_{ji}})v_{c(j)}V_j + \boldsymbol{b}_i\right)^n}{\lambda_{ii}} - \frac{\left(\sum_{j=1}^{N}(\boldsymbol{t_{ij}} - \boldsymbol{t_{ji}})v_{c(j)}V_j + \boldsymbol{b}_i\right)^{n-1}}{\lambda_{ii}}}{\Delta t\, \dot{u}^{n-\frac{1}{2}}} \right)$
18.	Update Numerator and Denominator parts of c^n	$\leftarrow c_{num}^n = c_{num}^n + u_i^n\, {}^1K_{ii}^n u_i^n, \quad c_{denom}^n = c_{denom}^n + u_i^n u_i^n,$
19.	Update Damping Coefficient c^n	$\leftarrow c^n = 2\sqrt{\frac{c_{num}^n}{c_{denom}^n}}, while\ c_{denom}^n > 0\ and\ c^n \leq 2$
20.	**for each** material point i in the body **do**	
21.	**if** 1st iteration then, **do**	
22.	Compute $\dot{u}^{\frac{1}{2}}$	$\leftarrow \dot{u}^{\frac{1}{2}} = \frac{\left(\sum_{j=1}^{N}(\boldsymbol{t_{ij}} - \boldsymbol{t_{ji}})v_{c(j)}V_j + \boldsymbol{b_i}\right)^1}{2\lambda_{ii}}$
23.	**else, do**	
24.	Compute $\dot{u}^{n+\frac{1}{2}}$	$\leftarrow \dot{u}^{n+\frac{1}{2}} = \frac{\left((2 - c^n\Delta t)\dot{u}^{n-\frac{1}{2}} + 2\Delta t \frac{\left(\sum_{j=1}^{N}(\boldsymbol{t_{ij}} - \boldsymbol{t_{ji}})v_{c(j)}V_j + \boldsymbol{b_i}\right)^n}{\lambda_{ii}} \right)}{(2 + c^n\Delta t)}$
25.	Compute u^{n+1}	$\leftarrow u^{n+1} = u^n + \Delta t \dot{u}^{n+\frac{1}{2}}$
26.	Store $\dot{u}^{n-\frac{1}{2}}$	$\leftarrow \dot{u}^{n-\frac{1}{2}} = \dot{u}^{n+\frac{1}{2}}$