```python
1  # -*- coding: utf-8 -*-
2  """
3  Created on Wed Aug 14 16:32:05 2019
4
5  @author: VACALDER
6  """
7
8  # -------------------------------------------------------------------------
9  # |       PROGRAM TO CHECK TIME DEPENDENT PROPERTIES EFFECTS
     ON STRUCTURES     |
10 # |
11 # |            version: 2.0.2
12 # |            Victor A Calderon
13 # |            PhD Student/ Research Assistant
14 # |            NC STATE UNIVERSITY
15 # |            2021 (c)
16 # |
17 # |
18 # -------------------------------------------------------------------------
19
20
21 # -------------------------------------------------------------------------
22 # |                              IMPORTS
23 # -------------------------------------------------------------------------
24
25 # import the os module
26 import pandas as pd
27 import time
28
29 start_time = time.time()
30 import os
31 from LibUnitsMUS import *
32 import Build_RC_Column
33 import Postprocessor_of_data
34 import openseespy.opensees as ops
35
36 # -------------------------------------------------------------------------
37 # | VARIABLES THAT CHANGE WITH TIME
```

```python
38  # ------------------------------------------------------------
    # -------------------
39  #
40  #
41  # *cover = Cover of concrete in cm
42  # *Tcorr = Time to corrosion in yrs
43  # *Time  = Different times that are being analyzed
44  # *wcr   = Water to cement ratio
45  # *dbi   = Initial longitudinal bar diameter
46  # *dti   = Initial transverse steel diameter
47
48
49  compressive_strength_concrete = 5 * ksi
50  yield_strength_long_steel = 60 * ksi
51  yield_strength_trans_steel = 60 * ksi
52  iShapeFactor = [4,6,8]
53  iCL = [0,5,10,15,20]
54  iCLt = [0]
55  pid = ops.getPID()
56  np = ops.getNP()
57  GM_Path = r'/home/vacalderon/Documents/MainshocksParallel_2
    .0.3/GroundMotion_Mainshock_Records'
58  GMListing = os.listdir(GM_Path)
59  rootdir = r'/home/vacalderon/Documents/MainshocksParallel_2
    .0.3'
60  iALR = [0.05, 0.10, 0.15, 0.20]  # [0.10] #
61  GMDB = pd.read_csv('mainshock_file_database.csv')
62  GeomDB = pd.read_csv('column_database.csv')
63  counter = 0
64  # ------------------------------------------------------------
    # -------------------
65  # |                              BATCH RUN
66  # ------------------------------------------------------------
    # -------------------
67
68  for column, Crow in GeomDB.iterrows():
69      D = float(Crow['column_diameter'])
70      dbi = float(Crow['long_bar_diameter'])
71      nbi = float(Crow['number_of_bars_longitudinal'])
72      dti = float(Crow['trans_bar_diameter'])
73      sti = float(Crow['spacing_trans_steel'])
74      rhol = float(Crow['rho_l'])
75      rhov = float(Crow['rho_v'])
```

```python
76          for shapefactor in iShapeFactor:
77              Height_of_Column = shapefactor * D
78              for ALR in iALR:
79
80                  Ag = 0.25 * math.pi * D ** 2
81                  AxialLoad = compressive_strength_concrete * Ag
    * ALR
82
83                  for GM, row in GMDB.iterrows():
84                      i = -1
85                      GM_fn = row['horizontal_1_filename']
86                      GM_dt = row['dt_horizontal1']
87                      GM_npt = row['npt_horizontal1']
88                      print('GM = ', GM_fn)
89                      GM_file = GM_Path + '/' + GM_fn
90                      for CL in iCL:
91                          for CLt in iCLt:
92                              if (counter % np) == pid:
93
94                                  datadir = rootdir + "/" + "
    data" + "/" + GM_fn + "/CL" + str(CL) + "/CLt" + str(CLt
    ) + "/D" + str(D) + "/SF" + str(
95                                      shapefactor) + "/ALR" +
    str(ALR) + "/RhoL" + str(rhol) + "/Rhov" + str(rhov)
96                                  if not os.path.exists(datadir
    ):
97                                      os.makedirs(datadir)
98
99                                  if yield_strength_long_steel
     == 60*ksi:
100                                     alpha = 0.0075
101                                 elif yield_strength_long_steel
     == 80*ksi:
102                                     alpha = 0.0075
103
104                                 dblc = ((1 - CL*0.01) ** 0.5
    ) * dbi
105                                 Build_RC_Column.
    Build_RC_Column(D, Height_of_Column,
    compressive_strength_concrete,
106
    yield_strength_long_steel, yield_strength_trans_steel,
107
```

```python
107             dbi, dti, CL, dblc, nbi, CLt, sti, datadir, AxialLoad,
108
         GM_file, GM_dt, GM_npt, ALR,alpha)
109
110                             with open(datadir + "/
    Conditions.out", 'w') as f:
111                                 f.write("%s \n" % (CL))
112                             f.close
113
114                             Postprocessor_of_data.
    Postprocessor_of_data(GM_fn, CL, CLt, D, shapefactor, ALR
    , rhol, rhov)
115
116                             os.remove(datadir + "/
    StressStrain.out")
117                             os.remove(datadir + "/
    StressStrain2.out")
118                             os.remove(datadir + "/
    StressStrain3.out")
119                             os.remove(datadir + "/
    StressStrain4.out")
120                             os.remove(datadir + "/
    Conditions.out")
121                             os.remove(datadir + "/DFree.
    out")
122                             os.remove(datadir + "/mat.out"
    )
123                             os.remove(datadir + "/Period.
    out")
124                             os.remove(datadir + "/PGA.out"
    )
125                             os.remove(datadir + "/RBase.
    out")
126                         counter += 1
127 print("ALL ANALYSIS COMPLETE")
128 print("--- %s minutes ---" % ((time.time() - start_time
    ) / 60))
129
```