# Model code for base case model

```
# Basic abridged network model for methane or hydrogen, version 3
# Western branch of the network has been removed, that is Kirriemuir -Stirling - Glasgow area - Gretna.
# Thsi includes pipes nos. PN07 - PN17 and PD32 - PD47.  These numbers are therefore not used here.
# PN17 was created to provide for export at the south end of the network at Coldstream, so is also not used
here, although it is #'d out.
# This version of the model only contains the East coast segments,
# chosen for simplicity because it has fewer nodes and no loop, essentially it's a straight line with several
branches.
# Updates over v2:  Varying demand profile through the day.  Introduce mixed gas as placeholder.

from datetime import datetime
# Getting the current date and time
dt = datetime.now().strftime("%Y-%m-%d %H:%M:%S")
print()
print()
print("started at ", dt)

### Define which gas is to be used (Methane,  Hydrogen, 80/20 mix by volume of Methane and hydrogen
respectively) ###
gas = "hydrogen"
#gas = "natural gas"
#gas = "80/20 mix"

# Create output file.  NB this will over-write previous versions, so go and save them under a different name
first if you want to keep them.
if gas == "hydrogen":
    outputfile = "East network model output base with linepack v3 (hydrogen).txt"
elif gas == "natural gas":
    outputfile = "East network model output base with linepack v3 (natural gas).txt"
elif gas == "80/20 mix":
    outputfile = "East network model output base with linepack v3 (80-20 mix).txt"
else:
    print("please check gas selection")

print("writing to ",outputfile)

with open(outputfile, "w") as fileout:
    print("Output file for abridged network model, loosely based on the eastern part of the transmission and
distribution network in Scotland", file=fileout)

    print("", file=fileout)
    print("", file=fileout)

    print("Started: ", dt, file=fileout)
    print("Gas type: ",gas, file=fileout)
    print("", file=fileout)

    import math
    #import numpy

    print("----- Section 1.  Basic data. ------", file=fileout)
    #
    # Define input and output node locations (national grid, metres)
    # [Reference, Easting, Northing, elevation mOD]  # elevation taken as constant initially as gravity effect
on gas is small.

    # Network node points
    NS00_Sfeg = ["St Fergus", 409589, 852173, 50]
    NN01_Petc = ["Peterculter", 384019, 800606, 50]
    NN02_Foch = ["Fochabers", 334763, 858643, 50]
    NN03_Kiri = ["Kirriemuir", 338549, 753933, 50]
    NN04_Glen = ["Glendoick", 314935, 722447, 50]
    NN05_Dunf = ["Dunfermline", 308970, 687449, 50]
    NN06_Balo = ["Balerno", 316345, 665962, 50]
    NN07_Laud = ["Lauder", 353041, 647613, 50]
    #NN16_Cold = ["Coldstream", 384180, 639767, 50]

    # Demand node locations
    ND26_Wlot = ["West Lothian", 305331, 666598, 50]
```

```
    ND27_Edin = ["Edinburgh", 325893, 674006, 50]
    ND28_Mlot = ["Midlothian", 333180, 667318, 50]
    ND29_Elot = ["East_Lothian", 351438, 673895, 50]
    ND31_Fife = ["Fife", 327397, 701016, 50]
    ND32_Dund = ["Dundee", 340311, 730250, 50]
    ND33_Angs = ["Angus", 345627, 750629, 50]
    ND34_Absh = ["Aberdeenshire", 369496, 795704, 50]
    ND35_Abdn = ["Aberdeen", 394483, 806377, 50]
    ND36_Mory = ["Moray", 321621, 862858, 50]
    ND37_High = ["Highland", 266642, 845382, 50]
    ND39_Pkin = ["Perth_and_Kinross", 311608, 723534, 50]
    ND46_Sbor = ["Scottish_Borders", 357711, 631683, 50]

    #Pipenames = [PN00_Sfeg_Petc, PN01_Petc_Foch, PN02_Petc_Kiri, PN03_Kiri_Glen, PN04_Glen_Dunf,
PN05_Dunf_Balo, PN06_Balo_Laud, PD27_Balo_Wlot, \
    #              PD28_Balo_Edin, PD29_Balo_Mlot, PD30_Laud_Elot, PD26_Dunf_Fife, PD24_Glen_Dund,
PD23_Kiri_Angs, PD20_Petc_Absh, PD19_Petc_Abdn, \
    #                 PD21_Foch_Mory, PD22_Foch_High, PD25_Glen_Pkin, PD31_Laud_Sbor]

    # Calculate pipe lengths (metres), assuming straight between nodes
    # Network pipes
    Len_PN00_Sfeg_Petc = ((NS00_Sfeg[1] - NN01_Petc[1])**2 + (NS00_Sfeg[2] - NN01_Petc[2])**2) **0.5
    Len_PN01_Petc_Foch = ((NN01_Petc[1] - NN02_Foch[1])**2 + (NN01_Petc[2] - NN02_Foch[2])**2) **0.5
    Len_PN02_Petc_Kiri = ((NN01_Petc[1] - NN03_Kiri[1])**2 + (NN01_Petc[2] - NN03_Kiri[2])**2) **0.5
    Len_PN03_Kiri_Glen = ((NN03_Kiri[1] - NN04_Glen[1])**2 + (NN03_Kiri[2] - NN04_Glen[2])**2) **0.5
    Len_PN04_Glen_Dunf = ((NN04_Glen[1] - NN05_Dunf[1])**2 + (NN04_Glen[2] - NN05_Dunf[2])**2) **0.5
    Len_PN05_Dunf_Balo = ((NN05_Dunf[1] - NN06_Balo[1])**2 + (NN05_Dunf[2] - NN06_Balo[2])**2) **0.5
    Len_PN06_Balo_Laud = ((NN06_Balo[1] - NN07_Laud[1])**2 + (NN06_Balo[2] - NN07_Laud[2])**2) **0.5
    #Len_PN07_Laud_Cold = ((NN07_Laud[1] - NN16_Cold[1])**2 + (NN07_Laud[2] - NN16_Cold[2])**2) **0.5
    #Demand_pipes
    Len_PD19_Petc_Abdn = ((NN01_Petc[1] - ND35_Abdn[1])**2 + (NN01_Petc[2] - ND35_Abdn[2])**2) **0.5
    Len_PD20_Petc_Absh = ((NN01_Petc[1] - ND34_Absh[1])**2 + (NN01_Petc[2] - ND34_Absh[2])**2) **0.5
    Len_PD21_Foch_Mory = ((NN02_Foch[1] - ND36_Mory[1])**2 + (NN02_Foch[2] - ND36_Mory[2])**2) **0.5
    Len_PD22_Foch_High = ((NN02_Foch[1] - ND37_High[1])**2 + (NN02_Foch[2] - ND37_High[2])**2) **0.5
    Len_PD23_Kiri_Angs = ((NN03_Kiri[1] - ND33_Angs[1])**2 + (NN03_Kiri[2] - ND33_Angs[2])**2) **0.5
    Len_PD24_Glen_Dund = ((NN04_Glen[1] - ND32_Dund[1])**2 + (NN04_Glen[2] - ND32_Dund[2])**2) **0.5
    Len_PD25_Glen_Pkin = ((NN04_Glen[1] - ND39_Pkin[1])**2 + (NN04_Glen[2] - ND39_Pkin[2])**2) **0.5
    Len_PD26_Dunf_Fife = ((NN05_Dunf[1] - ND31_Fife[1])**2 + (NN05_Dunf[2] - ND31_Fife[2])**2) **0.5
    Len_PD27_Balo_Wlot = ((NN06_Balo[1] - ND26_Wlot[1])**2 + (NN06_Balo[2] - ND26_Wlot[2])**2) **0.5
    Len_PD28_Balo_Edin = ((NN06_Balo[1] - ND27_Edin[1])**2 + (NN06_Balo[2] - ND27_Edin[2])**2) **0.5
    Len_PD29_Balo_Mlot = ((NN06_Balo[1] - ND28_Mlot[1])**2 + (NN06_Balo[2] - ND28_Mlot[2])**2) **0.5
    Len_PD30_Laud_Elot = ((NN07_Laud[1] - ND29_Elot[1])**2 + (NN07_Laud[2] - ND29_Elot[2])**2) **0.5
    Len_PD31_Laud_Sbor = ((NN07_Laud[1] - ND46_Sbor[1])**2 + (NN07_Laud[2] - ND46_Sbor[2])**2) **0.5

    print("",file=fileout)

    print("Pipe lengths (m):", file=fileout)
    print("Network_Pipes", file=fileout)
    print("Len_PN00_Sfeg_Petc: ",round(Len_PN00_Sfeg_Petc), file=fileout)
    print("Len_PN01_Petc_Foch: ",round(Len_PN01_Petc_Foch), file=fileout)
    print("Len_PN02_Petc_Kiri: ",round(Len_PN02_Petc_Kiri), file=fileout)
    print("Len_PN03_Kiri_Glen: ",round(Len_PN03_Kiri_Glen), file=fileout)
    print("Len_PN04_Glen_Dunf: ",round(Len_PN04_Glen_Dunf), file=fileout)
    print("Len_PN05_Dunf_Balo: ",round(Len_PN05_Dunf_Balo), file=fileout)
    print("Len_PN06_Balo_Laud: ",round(Len_PN06_Balo_Laud), file=fileout)
    #print("Len_PN07_Laud_Cold: ",round(Len_PN07_Laud_Cold), file=fileout)
    print("",file=fileout)

    print("Demand_pipes",file=fileout)
    print("Len_PD19_Petc_Abdn: ",round(Len_PD19_Petc_Abdn), file=fileout)
    print("Len_PD20_Petc_Absh: ",round(Len_PD20_Petc_Absh), file=fileout)
    print("Len_PD21_Foch_Mory: ",round(Len_PD21_Foch_Mory), file=fileout)
    print("Len_PD22_Foch_High: ",round(Len_PD22_Foch_High), file=fileout)
    print("Len_PD23_Kiri_Angs: ",round(Len_PD23_Kiri_Angs), file=fileout)
    print("Len_PD24_Glen_Dund: ",round(Len_PD24_Glen_Dund), file=fileout)
    print("Len_PD25_Glen_Pkin: ",round(Len_PD25_Glen_Pkin), file=fileout)
    print("Len_PD26_Dunf_Fife: ",round(Len_PD26_Dunf_Fife), file=fileout)
    print("Len_PD27_Balo_Wlot: ",round(Len_PD27_Balo_Wlot), file=fileout)
    print("Len_PD28_Balo_Edin: ",round(Len_PD28_Balo_Edin), file=fileout)
    print("Len_PD29_Balo_Mlot: ",round(Len_PD29_Balo_Mlot), file=fileout)
    print("Len_PD30_Laud_Elot: ",round(Len_PD30_Laud_Elot), file=fileout)
    print("Len_PD31_Laud_Sbor: ",round(Len_PD31_Laud_Sbor), file=fileout)
```

```
    # Define pipe parameters - diameters (m), roughness coefficients(mm), and minimum acceptable downstream
pressure (kPa).
    # Network pipes
    # These are set to the minimum standard size which allows service to be met with natural gas / methane
    #                                                  OG
    Param_PN00_Sfeg_Petc = [0.500, 0.0015, 0]  #fixed 500
    Param_PN01_Petc_Foch = [0.250, 0.0015, 0]  #fixed 250
    Param_PN02_Petc_Kiri = [0.450, 0.0015, 0]  #fixed 450
    Param_PN03_Kiri_Glen = [0.450, 0.0015, 0]  #fixed 450
    Param_PN04_Glen_Dunf = [0.450, 0.0015, 0]  #fixed 450
    Param_PN05_Dunf_Balo = [0.400, 0.0015, 0]  #fixed 400
    Param_PN06_Balo_Laud = [0.250, 0.0015, 0]  #fixed 250
    #Param_PN07_Laud_Cold = [0.050, 0.0015, 0]  #

    #Demand pipes
    #These are set to the minimum standard size which allows service to be met with natural gas / methane
    Param_PD19_Petc_Abdn = [0.350, 0.0015, 0] # fixed 350
    Param_PD20_Petc_Absh = [0.400, 0.0015, 0] # fixed 400
    Param_PD21_Foch_Mory = [0.250, 0.0015, 0] # fixed 250
    Param_PD22_Foch_High = [0.500, 0.0015, 0] # fixed 500
    Param_PD23_Kiri_Angs = [0.250, 0.0015, 0] # fixed 250
    Param_PD24_Glen_Dund = [0.350, 0.0015, 0] # fixed 350
    Param_PD25_Glen_Pkin = [0.250, 0.0015, 0] # fixed 250
    Param_PD26_Dunf_Fife = [0.450, 0.0015, 0] # fixed 450
    Param_PD27_Balo_Wlot = [0.300, 0.0015, 0] # fixed 300
    Param_PD28_Balo_Edin = [0.450, 0.0015, 0] # fixed 450
    Param_PD29_Balo_Mlot = [0.250, 0.0015, 0] # fixed 250
    Param_PD30_Laud_Elot = [0.300, 0.0015, 0] # fixed 300
    Param_PD31_Laud_Sbor = [0.300, 0.0015, 0] # fixed 300

     # Define density and viscosity for natural gas, given pressure
    #                          Pressure   Density        Viscosity
    #                            (kPa)    (kg/m3)         (uPa-s)
    Natural_gas_properties = {  10      :[     0.079    ,    11.073    ],
                                20      :[     0.158    ,    11.075    ],
                                30      :[     0.247    ,    11.078    ],
                                40      :[     0.348    ,    11.081    ],
                                50      :[     0.467    ,    11.085    ],
                                60      :[     0.609    ,    11.090    ],
                                70      :[     0.787    ,    11.095    ],
                                80      :[     1.024    ,    11.103    ],
                                90      :[     1.379    ,    11.114    ],
                                100     :[     0.790    ,    11.095    ],
                                110     :[     0.870    ,    11.098    ],
                                120     :[     0.950    ,    11.100    ],
                                130     :[     1.030    ,    11.103    ],
                                140     :[     1.110    ,    11.105    ],
                                150     :[     1.190    ,    11.108    ],
                                160     :[     1.270    ,    11.110    ],
                                170     :[     1.350    ,    11.113    ],
                                180     :[     1.430    ,    11.115    ],
                                190     :[     1.510    ,    11.118    ],
                                200     :[     1.590    ,    11.120    ],
                                210     :[     1.670    ,    11.122    ],
                                220     :[     1.750    ,    11.124    ],
                                230     :[     1.830    ,    11.126    ],
                                240     :[     1.910    ,    11.127    ],
                                250     :[     1.990    ,    11.129    ],
                                260     :[     2.070    ,    11.131    ],
                                270     :[     2.150    ,    11.133    ],
                                280     :[     2.230    ,    11.134    ],
                                290     :[     2.310    ,    11.136    ],
                                300     :[     2.390    ,    11.138    ],
                                310     :[     2.471    ,    11.139    ],
                                320     :[     2.552    ,    11.141    ],
                                330     :[     2.633    ,    11.142    ],
                                340     :[     2.714    ,    11.144    ],
                                350     :[     2.795    ,    11.145    ],
                                360     :[     2.876    ,    11.146    ],
                                370     :[     2.957    ,    11.148    ],
                                380     :[     3.038    ,    11.149    ],
                                390     :[     3.119    ,    11.151    ],
                                400     :[     3.200    ,    11.152    ],
                                410     :[     3.281    ,    11.153    ],
```

NB this table continues to page 231~~96~~. The corresponding hydrogen table that follows it, continues to page 243 08

```
    420     :[    3.362    ,    11.155    ],
    430     :[    3.443    ,    11.156    ],
    440     :[    3.524    ,    11.157    ],
    450     :[    3.605    ,    11.159    ],
    460     :[    3.686    ,    11.160    ],
    470     :[    3.767    ,    11.162    ],
    480     :[    3.848    ,    11.163    ],
    490     :[    3.929    ,    11.164    ],
    500     :[    4.010    ,    11.166    ],
    510     :[    4.091    ,    11.167    ],
    520     :[    4.172    ,    11.168    ],
    530     :[    4.253    ,    11.169    ],
    540     :[    4.334    ,    11.170    ],
    550     :[    4.415    ,    11.171    ],
    560     :[    4.496    ,    11.172    ],
    570     :[    4.577    ,    11.173    ],
    580     :[    4.658    ,    11.175    ],
    590     :[    4.739    ,    11.176    ],
    600     :[    4.820    ,    11.177    ],
    610     :[    4.901    ,    11.178    ],
    620     :[    4.982    ,    11.179    ],
    630     :[    5.063    ,    11.180    ],
    640     :[    5.144    ,    11.182    ],
    650     :[    5.225    ,    11.183    ],
    660     :[    5.306    ,    11.184    ],
    670     :[    5.387    ,    11.185    ],
    680     :[    5.468    ,    11.187    ],
    690     :[    5.549    ,    11.188    ],
    700     :[    5.630    ,    11.189    ],
    710     :[    5.712    ,    11.191    ],
    720     :[    5.794    ,    11.192    ],
    730     :[    5.876    ,    11.194    ],
    740     :[    5.958    ,    11.196    ],
    750     :[    6.040    ,    11.197    ],
    760     :[    6.122    ,    11.199    ],
    770     :[    6.204    ,    11.200    ],
    780     :[    6.286    ,    11.202    ],
    790     :[    6.368    ,    11.204    ],
    800     :[    6.450    ,    11.205    ],
    810     :[    6.532    ,    11.207    ],
    820     :[    6.614    ,    11.208    ],
    830     :[    6.696    ,    11.210    ],
    840     :[    6.778    ,    11.212    ],
    850     :[    6.860    ,    11.213    ],
    860     :[    6.942    ,    11.215    ],
    870     :[    7.024    ,    11.217    ],
    880     :[    7.106    ,    11.218    ],
    890     :[    7.188    ,    11.220    ],
    900     :[    7.270    ,    11.221    ],
    910     :[    7.353    ,    11.223    ],
    920     :[    7.436    ,    11.225    ],
    930     :[    7.519    ,    11.226    ],
    940     :[    7.602    ,    11.228    ],
    950     :[    7.685    ,    11.230    ],
    960     :[    7.768    ,    11.231    ],
    970     :[    7.851    ,    11.233    ],
    980     :[    7.934    ,    11.235    ],
    990     :[    8.017    ,    11.236    ],
   1000     :[    8.100    ,    11.238    ],
   1010     :[    8.185    ,    11.240    ],
   1020     :[    8.270    ,    11.241    ],
   1030     :[    8.355    ,    11.243    ],
   1040     :[    8.440    ,    11.245    ],
   1050     :[    8.525    ,    11.247    ],
   1060     :[    8.610    ,    11.249    ],
   1070     :[    8.695    ,    11.251    ],
   1080     :[    8.780    ,    11.252    ],
   1090     :[    8.865    ,    11.254    ],
   1100     :[    8.950    ,    11.256    ],
   1110     :[    9.035    ,    11.258    ],
   1120     :[    9.120    ,    11.260    ],
   1130     :[    9.205    ,    11.262    ],
   1140     :[    9.290    ,    11.263    ],
   1150     :[    9.375    ,    11.265    ],
```

```
                1160    :[     9.460    ,    11.267    ],
                1170    :[     9.545    ,    11.269    ],
                1180    :[     9.630    ,    11.271    ],
                1190    :[     9.715    ,    11.272    ],
                1200    :[     9.800    ,    11.274    ],
                1210    :[     9.885    ,    11.276    ],
                1220    :[     9.970    ,    11.278    ],
                1230    :[    10.055    ,    11.280    ],
                1240    :[    10.140    ,    11.282    ],
                1250    :[    10.225    ,    11.283    ],
                1260    :[    10.310    ,    11.285    ],
                1270    :[    10.395    ,    11.287    ],
                1280    :[    10.480    ,    11.289    ],
                1290    :[    10.565    ,    11.291    ],
                1300    :[    10.650    ,    11.293    ],
                1310    :[    10.735    ,    11.294    ],
                1320    :[    10.820    ,    11.296    ],
                1330    :[    10.905    ,    11.298    ],
                1340    :[    10.990    ,    11.300    ],
                1350    :[    11.075    ,    11.302    ],
                1360    :[    11.160    ,    11.304    ],
                1370    :[    11.245    ,    11.305    ],
                1380    :[    11.330    ,    11.307    ],
                1390    :[    11.415    ,    11.309    ],
                1400    :[    11.500    ,    11.311    ],
                1410    :[    11.585    ,    11.313    ],
                1420    :[    11.670    ,    11.314    ],
                1430    :[    11.755    ,    11.316    ],
                1440    :[    11.840    ,    11.318    ],
                1450    :[    11.925    ,    11.320    ],
                1460    :[    12.010    ,    11.322    ],
                1470    :[    12.095    ,    11.324    ],
                1480    :[    12.180    ,    11.325    ],
                1490    :[    12.265    ,    11.327    ],
                1500    :[    12.350    ,    11.329    ],
                1510    :[    12.435    ,    11.331    ],
                1520    :[    12.520    ,    11.333    ],
                1530    :[    12.605    ,    11.335    ],
                1540    :[    12.690    ,    11.336    ],
                1550    :[    12.775    ,    11.338    ],
                1560    :[    12.860    ,    11.340    ],
                1570    :[    12.945    ,    11.342    ],
                1580    :[    13.030    ,    11.344    ],
                1590    :[    13.115    ,    11.346    ],
                1600    :[    13.200    ,    11.347    ],
                1610    :[    13.285    ,    11.349    ],
                1620    :[    13.370    ,    11.351    ],
                1630    :[    13.455    ,    11.353    ],
                1640    :[    13.540    ,    11.355    ],
                1650    :[    13.625    ,    11.356    ],
                1660    :[    13.710    ,    11.358    ],
                1670    :[    13.795    ,    11.360    ],
                1680    :[    13.880    ,    11.362    ],
                1690    :[    13.965    ,    11.364    ],
                1700    :[    14.050    ,    11.366    ],
                1710    :[    14.135    ,    11.367    ],
                1720    :[    14.220    ,    11.369    ],
                1730    :[    14.305    ,    11.371    ],
                1740    :[    14.390    ,    11.373    ],
                1750    :[    14.475    ,    11.375    ],
                1760    :[    14.560    ,    11.377    ],
                1770    :[    14.645    ,    11.378    ],
                1780    :[    14.730    ,    11.380    ],
                1790    :[    14.815    ,    11.382    ],
                1800    :[    14.900    ,    11.384    ],
                1810    :[    14.985    ,    11.386    ],
                1820    :[    15.070    ,    11.387    ],
                1830    :[    15.155    ,    11.389    ],
                1840    :[    15.240    ,    11.391    ],
                1850    :[    15.325    ,    11.393    ],
                1860    :[    15.410    ,    11.395    ],
                1870    :[    15.495    ,    11.397    ],
                1880    :[    15.580    ,    11.398    ],
                1890    :[    15.665    ,    11.400    ],
```

```
                1900    :[    15.750    ,    11.402    ],
                1910    :[    15.835    ,    11.404    ],
                1920    :[    15.920    ,    11.406    ],
                1930    :[    16.005    ,    11.408    ],
                1940    :[    16.090    ,    11.409    ],
                1950    :[    16.175    ,    11.411    ],
                1960    :[    16.260    ,    11.413    ],
                1970    :[    16.345    ,    11.415    ],
                1980    :[    16.430    ,    11.417    ],
                1990    :[    16.515    ,    11.419    ],
                2000    :[    16.600    ,    11.420    ],
                2010    :[    16.688    ,    11.422    ],
                2020    :[    16.776    ,    11.425    ],
                2030    :[    16.864    ,    11.427    ],
                2040    :[    16.952    ,    11.429    ],
                2050    :[    17.040    ,    11.431    ],
                2060    :[    17.128    ,    11.433    ],
                2070    :[    17.216    ,    11.435    ],
                2080    :[    17.304    ,    11.437    ],
                2090    :[    17.392    ,    11.439    ],
                2100    :[    17.480    ,    11.441    ],
                2110    :[    17.568    ,    11.444    ],
                2120    :[    17.656    ,    11.446    ],
                2130    :[    17.744    ,    11.448    ],
                2140    :[    17.832    ,    11.450    ],
                2150    :[    17.920    ,    11.452    ],
                2160    :[    18.008    ,    11.454    ],
                2170    :[    18.096    ,    11.456    ],
                2180    :[    18.184    ,    11.458    ],
                2190    :[    18.272    ,    11.460    ],
                2200    :[    18.360    ,    11.462    ],
                2210    :[    18.448    ,    11.465    ],
                2220    :[    18.536    ,    11.467    ],
                2230    :[    18.624    ,    11.469    ],
                2240    :[    18.712    ,    11.471    ],
                2250    :[    18.800    ,    11.473    ],
                2260    :[    18.888    ,    11.475    ],
                2270    :[    18.976    ,    11.477    ],
                2280    :[    19.064    ,    11.479    ],
                2290    :[    19.152    ,    11.481    ],
                2300    :[    19.240    ,    11.484    ],
                2310    :[    19.328    ,    11.486    ],
                2320    :[    19.416    ,    11.488    ],
                2330    :[    19.504    ,    11.490    ],
                2340    :[    19.592    ,    11.492    ],
                2350    :[    19.680    ,    11.494    ],
                2360    :[    19.768    ,    11.496    ],
                2370    :[    19.856    ,    11.498    ],
                2380    :[    19.944    ,    11.500    ],
                2390    :[    20.032    ,    11.502    ],
                2400    :[    20.120    ,    11.505    ],
                2410    :[    20.208    ,    11.507    ],
                2420    :[    20.296    ,    11.509    ],
                2430    :[    20.384    ,    11.511    ],
                2440    :[    20.472    ,    11.513    ],
                2450    :[    20.560    ,    11.515    ],
                2460    :[    20.648    ,    11.517    ],
                2470    :[    20.736    ,    11.519    ],
                2480    :[    20.824    ,    11.521    ],
                2490    :[    20.912    ,    11.524    ],
                2500    :[    21.000    ,    11.526    ],
                2510    :[    21.088    ,    11.528    ],
                2520    :[    21.176    ,    11.530    ],
                2530    :[    21.264    ,    11.532    ],
                2540    :[    21.352    ,    11.534    ],
                2550    :[    21.440    ,    11.536    ],
                2560    :[    21.528    ,    11.538    ],
                2570    :[    21.616    ,    11.540    ],
                2580    :[    21.704    ,    11.543    ],
                2590    :[    21.792    ,    11.545    ],
                2600    :[    21.880    ,    11.547    ],
                2610    :[    21.968    ,    11.549    ],
                2620    :[    22.056    ,    11.551    ],
                2630    :[    22.144    ,    11.553    ],
```

```
    2640    :[    22.232    ,    11.555    ],
    2650    :[    22.320    ,    11.557    ],
    2660    :[    22.408    ,    11.559    ],
    2670    :[    22.496    ,    11.561    ],
    2680    :[    22.584    ,    11.564    ],
    2690    :[    22.672    ,    11.566    ],
    2700    :[    22.760    ,    11.568    ],
    2710    :[    22.848    ,    11.570    ],
    2720    :[    22.936    ,    11.572    ],
    2730    :[    23.024    ,    11.574    ],
    2740    :[    23.112    ,    11.576    ],
    2750    :[    23.200    ,    11.578    ],
    2760    :[    23.288    ,    11.580    ],
    2770    :[    23.376    ,    11.583    ],
    2780    :[    23.464    ,    11.585    ],
    2790    :[    23.552    ,    11.587    ],
    2800    :[    23.640    ,    11.589    ],
    2810    :[    23.728    ,    11.591    ],
    2820    :[    23.816    ,    11.593    ],
    2830    :[    23.904    ,    11.595    ],
    2840    :[    23.992    ,    11.597    ],
    2850    :[    24.080    ,    11.599    ],
    2860    :[    24.168    ,    11.602    ],
    2870    :[    24.256    ,    11.604    ],
    2880    :[    24.344    ,    11.606    ],
    2890    :[    24.432    ,    11.608    ],
    2900    :[    24.520    ,    11.610    ],
    2910    :[    24.608    ,    11.612    ],
    2920    :[    24.696    ,    11.614    ],
    2930    :[    24.784    ,    11.616    ],
    2940    :[    24.872    ,    11.618    ],
    2950    :[    24.960    ,    11.620    ],
    2960    :[    25.048    ,    11.623    ],
    2970    :[    25.136    ,    11.625    ],
    2980    :[    25.224    ,    11.627    ],
    2990    :[    25.312    ,    11.629    ],
    3000    :[    25.400    ,    11.631    ],
    3010    :[    25.493    ,    11.634    ],
    3020    :[    25.586    ,    11.636    ],
    3030    :[    25.679    ,    11.639    ],
    3040    :[    25.772    ,    11.642    ],
    3050    :[    25.865    ,    11.644    ],
    3060    :[    25.958    ,    11.647    ],
    3070    :[    26.051    ,    11.650    ],
    3080    :[    26.144    ,    11.652    ],
    3090    :[    26.237    ,    11.655    ],
    3100    :[    26.330    ,    11.658    ],
    3110    :[    26.423    ,    11.660    ],
    3120    :[    26.516    ,    11.663    ],
    3130    :[    26.609    ,    11.666    ],
    3140    :[    26.702    ,    11.668    ],
    3150    :[    26.795    ,    11.671    ],
    3160    :[    26.888    ,    11.674    ],
    3170    :[    26.981    ,    11.676    ],
    3180    :[    27.074    ,    11.679    ],
    3190    :[    27.167    ,    11.681    ],
    3200    :[    27.260    ,    11.684    ],
    3210    :[    27.353    ,    11.687    ],
    3220    :[    27.446    ,    11.689    ],
    3230    :[    27.539    ,    11.692    ],
    3240    :[    27.632    ,    11.695    ],
    3250    :[    27.725    ,    11.697    ],
    3260    :[    27.818    ,    11.700    ],
    3270    :[    27.911    ,    11.703    ],
    3280    :[    28.004    ,    11.705    ],
    3290    :[    28.097    ,    11.708    ],
    3300    :[    28.190    ,    11.711    ],
    3310    :[    28.283    ,    11.713    ],
    3320    :[    28.376    ,    11.716    ],
    3330    :[    28.469    ,    11.719    ],
    3340    :[    28.562    ,    11.721    ],
    3350    :[    28.655    ,    11.724    ],
    3360    :[    28.748    ,    11.727    ],
    3370    :[    28.841    ,    11.729    ],
```

```
    3380    :[    28.934    ,    11.732    ],
    3390    :[    29.027    ,    11.735    ],
    3400    :[    29.120    ,    11.737    ],
    3410    :[    29.213    ,    11.740    ],
    3420    :[    29.306    ,    11.743    ],
    3430    :[    29.399    ,    11.745    ],
    3440    :[    29.492    ,    11.748    ],
    3450    :[    29.585    ,    11.751    ],
    3460    :[    29.678    ,    11.753    ],
    3470    :[    29.771    ,    11.756    ],
    3480    :[    29.864    ,    11.759    ],
    3490    :[    29.957    ,    11.761    ],
    3500    :[    30.050    ,    11.764    ],
    3510    :[    30.143    ,    11.767    ],
    3520    :[    30.236    ,    11.769    ],
    3530    :[    30.329    ,    11.772    ],
    3540    :[    30.422    ,    11.774    ],
    3550    :[    30.515    ,    11.777    ],
    3560    :[    30.608    ,    11.780    ],
    3570    :[    30.701    ,    11.782    ],
    3580    :[    30.794    ,    11.785    ],
    3590    :[    30.887    ,    11.788    ],
    3600    :[    30.980    ,    11.790    ],
    3610    :[    31.073    ,    11.793    ],
    3620    :[    31.166    ,    11.796    ],
    3630    :[    31.259    ,    11.798    ],
    3640    :[    31.352    ,    11.801    ],
    3650    :[    31.445    ,    11.804    ],
    3660    :[    31.538    ,    11.806    ],
    3670    :[    31.631    ,    11.809    ],
    3680    :[    31.724    ,    11.812    ],
    3690    :[    31.817    ,    11.814    ],
    3700    :[    31.910    ,    11.817    ],
    3710    :[    32.003    ,    11.820    ],
    3720    :[    32.096    ,    11.822    ],
    3730    :[    32.189    ,    11.825    ],
    3740    :[    32.282    ,    11.828    ],
    3750    :[    32.375    ,    11.830    ],
    3760    :[    32.468    ,    11.833    ],
    3770    :[    32.561    ,    11.836    ],
    3780    :[    32.654    ,    11.838    ],
    3790    :[    32.747    ,    11.841    ],
    3800    :[    32.840    ,    11.844    ],
    3810    :[    32.933    ,    11.846    ],
    3820    :[    33.026    ,    11.849    ],
    3830    :[    33.119    ,    11.852    ],
    3840    :[    33.212    ,    11.854    ],
    3850    :[    33.305    ,    11.857    ],
    3860    :[    33.398    ,    11.860    ],
    3870    :[    33.491    ,    11.862    ],
    3880    :[    33.584    ,    11.865    ],
    3890    :[    33.677    ,    11.867    ],
    3900    :[    33.770    ,    11.870    ],
    3910    :[    33.863    ,    11.873    ],
    3920    :[    33.956    ,    11.875    ],
    3930    :[    34.049    ,    11.878    ],
    3940    :[    34.142    ,    11.881    ],
    3950    :[    34.235    ,    11.883    ],
    3960    :[    34.328    ,    11.886    ],
    3970    :[    34.421    ,    11.889    ],
    3980    :[    34.514    ,    11.891    ],
    3990    :[    34.607    ,    11.894    ],
    4000    :[    34.700    ,    11.897    ],
    4010    :[    34.797    ,    11.900    ],
    4020    :[    34.894    ,    11.903    ],
    4030    :[    34.991    ,    11.906    ],
    4040    :[    35.088    ,    11.909    ],
    4050    :[    35.185    ,    11.912    ],
    4060    :[    35.282    ,    11.915    ],
    4070    :[    35.379    ,    11.918    ],
    4080    :[    35.476    ,    11.921    ],
    4090    :[    35.573    ,    11.924    ],
    4100    :[    35.670    ,    11.927    ],
    4110    :[    35.767    ,    11.930    ],
```

```
    4120    :[    35.864    ,    11.933    ],
    4130    :[    35.961    ,    11.936    ],
    4140    :[    36.058    ,    11.939    ],
    4150    :[    36.155    ,    11.943    ],
    4160    :[    36.252    ,    11.946    ],
    4170    :[    36.349    ,    11.949    ],
    4180    :[    36.446    ,    11.952    ],
    4190    :[    36.543    ,    11.955    ],
    4200    :[    36.640    ,    11.958    ],
    4210    :[    36.737    ,    11.961    ],
    4220    :[    36.834    ,    11.964    ],
    4230    :[    36.931    ,    11.967    ],
    4240    :[    37.028    ,    11.970    ],
    4250    :[    37.125    ,    11.973    ],
    4260    :[    37.222    ,    11.976    ],
    4270    :[    37.319    ,    11.979    ],
    4280    :[    37.416    ,    11.982    ],
    4290    :[    37.513    ,    11.985    ],
    4300    :[    37.610    ,    11.988    ],
    4310    :[    37.707    ,    11.991    ],
    4320    :[    37.804    ,    11.995    ],
    4330    :[    37.901    ,    11.998    ],
    4340    :[    37.998    ,    12.001    ],
    4350    :[    38.095    ,    12.004    ],
    4360    :[    38.192    ,    12.007    ],
    4370    :[    38.289    ,    12.010    ],
    4380    :[    38.386    ,    12.013    ],
    4390    :[    38.483    ,    12.016    ],
    4400    :[    38.580    ,    12.019    ],
    4410    :[    38.677    ,    12.022    ],
    4420    :[    38.774    ,    12.025    ],
    4430    :[    38.871    ,    12.028    ],
    4440    :[    38.968    ,    12.031    ],
    4450    :[    39.065    ,    12.034    ],
    4460    :[    39.162    ,    12.037    ],
    4470    :[    39.259    ,    12.040    ],
    4480    :[    39.356    ,    12.043    ],
    4490    :[    39.453    ,    12.046    ],
    4500    :[    39.550    ,    12.050    ],
    4510    :[    39.647    ,    12.053    ],
    4520    :[    39.744    ,    12.056    ],
    4530    :[    39.841    ,    12.059    ],
    4540    :[    39.938    ,    12.062    ],
    4550    :[    40.035    ,    12.065    ],
    4560    :[    40.132    ,    12.068    ],
    4570    :[    40.229    ,    12.071    ],
    4580    :[    40.326    ,    12.074    ],
    4590    :[    40.423    ,    12.077    ],
    4600    :[    40.520    ,    12.080    ],
    4610    :[    40.617    ,    12.083    ],
    4620    :[    40.714    ,    12.086    ],
    4630    :[    40.811    ,    12.089    ],
    4640    :[    40.908    ,    12.092    ],
    4650    :[    41.005    ,    12.095    ],
    4660    :[    41.102    ,    12.098    ],
    4670    :[    41.199    ,    12.101    ],
    4680    :[    41.296    ,    12.105    ],
    4690    :[    41.393    ,    12.108    ],
    4700    :[    41.490    ,    12.111    ],
    4710    :[    41.587    ,    12.114    ],
    4720    :[    41.684    ,    12.117    ],
    4730    :[    41.781    ,    12.120    ],
    4740    :[    41.878    ,    12.123    ],
    4750    :[    41.975    ,    12.126    ],
    4760    :[    42.072    ,    12.129    ],
    4770    :[    42.169    ,    12.132    ],
    4780    :[    42.266    ,    12.135    ],
    4790    :[    42.363    ,    12.138    ],
    4800    :[    42.460    ,    12.141    ],
    4810    :[    42.557    ,    12.144    ],
    4820    :[    42.654    ,    12.147    ],
    4830    :[    42.751    ,    12.150    ],
    4840    :[    42.848    ,    12.153    ],
    4850    :[    42.945    ,    12.156    ],
```

```
        4860    :[    43.042    ,    12.160    ],
        4870    :[    43.139    ,    12.163    ],
        4880    :[    43.236    ,    12.166    ],
        4890    :[    43.333    ,    12.169    ],
        4900    :[    43.430    ,    12.172    ],
        4910    :[    43.527    ,    12.175    ],
        4920    :[    43.624    ,    12.178    ],
        4930    :[    43.721    ,    12.181    ],
        4940    :[    43.818    ,    12.184    ],
        4950    :[    43.915    ,    12.187    ],
        4960    :[    44.012    ,    12.190    ],
        4970    :[    44.109    ,    12.193    ],
        4980    :[    44.206    ,    12.196    ],
        4990    :[    44.303    ,    12.199    ],
        5000    :[    44.400    ,    12.202    ],
        5010    :[    44.501    ,    12.206    ],
        5020    :[    44.602    ,    12.209    ],
        5030    :[    44.703    ,    12.213    ],
        5040    :[    44.804    ,    12.216    ],
        5050    :[    44.905    ,    12.220    ],
        5060    :[    45.006    ,    12.223    ],
        5070    :[    45.107    ,    12.227    ],
        5080    :[    45.208    ,    12.230    ],
        5090    :[    45.309    ,    12.234    ],
        5100    :[    45.410    ,    12.237    ],
        5110    :[    45.511    ,    12.241    ],
        5120    :[    45.612    ,    12.244    ],
        5130    :[    45.713    ,    12.248    ],
        5140    :[    45.814    ,    12.251    ],
        5150    :[    45.915    ,    12.255    ],
        5160    :[    46.016    ,    12.259    ],
        5170    :[    46.117    ,    12.262    ],
        5180    :[    46.218    ,    12.266    ],
        5190    :[    46.319    ,    12.269    ],
        5200    :[    46.420    ,    12.273    ],
        5210    :[    46.521    ,    12.276    ],
        5220    :[    46.622    ,    12.280    ],
        5230    :[    46.723    ,    12.283    ],
        5240    :[    46.824    ,    12.287    ],
        5250    :[    46.925    ,    12.290    ],
        5260    :[    47.026    ,    12.294    ],
        5270    :[    47.127    ,    12.297    ],
        5280    :[    47.228    ,    12.301    ],
        5290    :[    47.329    ,    12.304    ],
        5300    :[    47.430    ,    12.308    ],
        5310    :[    47.531    ,    12.311    ],
        5320    :[    47.632    ,    12.315    ],
        5330    :[    47.733    ,    12.318    ],
        5340    :[    47.834    ,    12.322    ],
        5350    :[    47.935    ,    12.325    ],
        5360    :[    48.036    ,    12.329    ],
        5370    :[    48.137    ,    12.332    ],
        5380    :[    48.238    ,    12.336    ],
        5390    :[    48.339    ,    12.339    ],
        5400    :[    48.440    ,    12.343    ],
        5410    :[    48.541    ,    12.346    ],
        5420    :[    48.642    ,    12.350    ],
        5430    :[    48.743    ,    12.353    ],
        5440    :[    48.844    ,    12.357    ],
        5450    :[    48.945    ,    12.360    ],
        5460    :[    49.046    ,    12.364    ],
        5470    :[    49.147    ,    12.367    ],
        5480    :[    49.248    ,    12.371    ],
        5490    :[    49.349    ,    12.374    ],
        5500    :[    49.450    ,    12.378    ],
        5510    :[    49.551    ,    12.381    ],
        5520    :[    49.652    ,    12.385    ],
        5530    :[    49.753    ,    12.388    ],
        5540    :[    49.854    ,    12.392    ],
        5550    :[    49.955    ,    12.395    ],
        5560    :[    50.056    ,    12.399    ],
        5570    :[    50.157    ,    12.402    ],
        5580    :[    50.258    ,    12.406    ],
        5590    :[    50.359    ,    12.409    ],
```

```
        5600    :[    50.460    ,    12.413    ],
        5610    :[    50.561    ,    12.416    ],
        5620    :[    50.662    ,    12.420    ],
        5630    :[    50.763    ,    12.423    ],
        5640    :[    50.864    ,    12.427    ],
        5650    :[    50.965    ,    12.431    ],
        5660    :[    51.066    ,    12.434    ],
        5670    :[    51.167    ,    12.438    ],
        5680    :[    51.268    ,    12.441    ],
        5690    :[    51.369    ,    12.445    ],
        5700    :[    51.470    ,    12.448    ],
        5710    :[    51.571    ,    12.452    ],
        5720    :[    51.672    ,    12.455    ],
        5730    :[    51.773    ,    12.459    ],
        5740    :[    51.874    ,    12.462    ],
        5750    :[    51.975    ,    12.466    ],
        5760    :[    52.076    ,    12.469    ],
        5770    :[    52.177    ,    12.473    ],
        5780    :[    52.278    ,    12.476    ],
        5790    :[    52.379    ,    12.480    ],
        5800    :[    52.480    ,    12.483    ],
        5810    :[    52.581    ,    12.487    ],
        5820    :[    52.682    ,    12.490    ],
        5830    :[    52.783    ,    12.494    ],
        5840    :[    52.884    ,    12.497    ],
        5850    :[    52.985    ,    12.501    ],
        5860    :[    53.086    ,    12.504    ],
        5870    :[    53.187    ,    12.508    ],
        5880    :[    53.288    ,    12.511    ],
        5890    :[    53.389    ,    12.515    ],
        5900    :[    53.490    ,    12.518    ],
        5910    :[    53.591    ,    12.522    ],
        5920    :[    53.692    ,    12.525    ],
        5930    :[    53.793    ,    12.529    ],
        5940    :[    53.894    ,    12.532    ],
        5950    :[    53.995    ,    12.536    ],
        5960    :[    54.096    ,    12.539    ],
        5970    :[    54.197    ,    12.543    ],
        5980    :[    54.298    ,    12.546    ],
        5990    :[    54.399    ,    12.550    ],
        6000    :[    54.500    ,    12.553    ],
        6010    :[    54.606    ,    12.557    ],
        6020    :[    54.711    ,    12.561    ],
        6030    :[    54.817    ,    12.565    ],
        6040    :[    54.922    ,    12.570    ],
        6050    :[    55.028    ,    12.574    ],
        6060    :[    55.134    ,    12.578    ],
        6070    :[    55.239    ,    12.582    ],
        6080    :[    55.345    ,    12.586    ],
        6090    :[    55.450    ,    12.590    ],
        6100    :[    55.556    ,    12.594    ],
        6110    :[    55.662    ,    12.598    ],
        6120    :[    55.767    ,    12.602    ],
        6130    :[    55.873    ,    12.606    ],
        6140    :[    55.978    ,    12.610    ],
        6150    :[    56.084    ,    12.614    ],
        6160    :[    56.190    ,    12.618    ],
        6170    :[    56.295    ,    12.622    ],
        6180    :[    56.401    ,    12.626    ],
        6190    :[    56.506    ,    12.630    ],
        6200    :[    56.612    ,    12.634    ],
        6210    :[    56.718    ,    12.638    ],
        6220    :[    56.823    ,    12.642    ],
        6230    :[    56.929    ,    12.646    ],
        6240    :[    57.034    ,    12.650    ],
        6250    :[    57.140    ,    12.654    ],
        6260    :[    57.246    ,    12.658    ],
        6270    :[    57.351    ,    12.662    ],
        6280    :[    57.457    ,    12.666    ],
        6290    :[    57.562    ,    12.670    ],
        6300    :[    57.668    ,    12.674    ],
        6310    :[    57.774    ,    12.679    ],
        6320    :[    57.879    ,    12.683    ],
        6330    :[    57.985    ,    12.687    ],
```

```
        6340    :[    58.090    ,    12.691    ],
        6350    :[    58.196    ,    12.695    ],
        6360    :[    58.302    ,    12.699    ],
        6370    :[    58.407    ,    12.703    ],
        6380    :[    58.513    ,    12.707    ],
        6390    :[    58.618    ,    12.711    ],
        6400    :[    58.724    ,    12.715    ],
        6410    :[    58.830    ,    12.719    ],
        6420    :[    58.935    ,    12.723    ],
        6430    :[    59.041    ,    12.727    ],
        6440    :[    59.146    ,    12.731    ],
        6450    :[    59.252    ,    12.735    ],
        6460    :[    59.358    ,    12.739    ],
        6470    :[    59.463    ,    12.743    ],
        6480    :[    59.569    ,    12.747    ],
        6490    :[    59.674    ,    12.751    ],
        6500    :[    59.780    ,    12.755    ],
        6510    :[    59.886    ,    12.759    ],
        6520    :[    59.991    ,    12.763    ],
        6530    :[    60.097    ,    12.767    ],
        6540    :[    60.202    ,    12.771    ],
        6550    :[    60.308    ,    12.775    ],
        6560    :[    60.414    ,    12.779    ],
        6570    :[    60.519    ,    12.783    ],
        6580    :[    60.625    ,    12.787    ],
        6590    :[    60.730    ,    12.792    ],
        6600    :[    60.836    ,    12.796    ],
        6610    :[    60.942    ,    12.800    ],
        6620    :[    61.047    ,    12.804    ],
        6630    :[    61.153    ,    12.808    ],
        6640    :[    61.258    ,    12.812    ],
        6650    :[    61.364    ,    12.816    ],
        6660    :[    61.470    ,    12.820    ],
        6670    :[    61.575    ,    12.824    ],
        6680    :[    61.681    ,    12.828    ],
        6690    :[    61.786    ,    12.832    ],
        6700    :[    61.892    ,    12.836    ],
        6710    :[    61.998    ,    12.840    ],
        6720    :[    62.103    ,    12.844    ],
        6730    :[    62.209    ,    12.848    ],
        6740    :[    62.314    ,    12.852    ],
        6750    :[    62.420    ,    12.856    ],
        6760    :[    62.526    ,    12.860    ],
        6770    :[    62.631    ,    12.864    ],
        6780    :[    62.737    ,    12.868    ],
        6790    :[    62.842    ,    12.872    ],
        6800    :[    62.948    ,    12.876    ],
        6810    :[    63.054    ,    12.880    ],
        6820    :[    63.159    ,    12.884    ],
        6830    :[    63.265    ,    12.888    ],
        6840    :[    63.370    ,    12.892    ],
        6850    :[    63.476    ,    12.896    ],
        6860    :[    63.582    ,    12.901    ],
        6870    :[    63.687    ,    12.905    ],
        6880    :[    63.793    ,    12.909    ],
        6890    :[    63.898    ,    12.913    ],
        6900    :[    64.004    ,    12.917    ],
        6910    :[    64.110    ,    12.921    ],
        6920    :[    64.215    ,    12.925    ],
        6930    :[    64.321    ,    12.929    ],
        6940    :[    64.426    ,    12.933    ],
        6950    :[    64.532    ,    12.937    ],
        6960    :[    64.638    ,    12.941    ],
        6970    :[    64.743    ,    12.945    ],
        6980    :[    64.849    ,    12.949    ],
        6990    :[    64.954    ,    12.953    ],
        7000    :[    65.060    ,    12.957    ],
        7010    :[    65.169    ,    12.962    ],
        7020    :[    65.278    ,    12.966    ],
        7030    :[    65.386    ,    12.971    ],
        7040    :[    65.495    ,    12.975    ],
        7050    :[    65.604    ,    12.980    ],
        7060    :[    65.713    ,    12.984    ],
        7070    :[    65.822    ,    12.989    ],
```

```
        7080    :[    65.930    ,    12.994    ],
        7090    :[    66.039    ,    12.998    ],
        7100    :[    66.148    ,    13.003    ],
        7110    :[    66.257    ,    13.007    ],
        7120    :[    66.366    ,    13.012    ],
        7130    :[    66.474    ,    13.017    ],
        7140    :[    66.583    ,    13.021    ],
        7150    :[    66.692    ,    13.026    ],
        7160    :[    66.801    ,    13.030    ],
        7170    :[    66.910    ,    13.035    ],
        7180    :[    67.018    ,    13.039    ],
        7190    :[    67.127    ,    13.044    ],
        7200    :[    67.236    ,    13.049    ],
        7210    :[    67.345    ,    13.053    ],
        7220    :[    67.454    ,    13.058    ],
        7230    :[    67.562    ,    13.062    ],
        7240    :[    67.671    ,    13.067    ],
        7250    :[    67.780    ,    13.071    ],
        7260    :[    67.889    ,    13.076    ],
        7270    :[    67.998    ,    13.081    ],
        7280    :[    68.106    ,    13.085    ],
        7290    :[    68.215    ,    13.090    ],
        7300    :[    68.324    ,    13.094    ],
        7310    :[    68.433    ,    13.099    ],
        7320    :[    68.542    ,    13.104    ],
        7330    :[    68.650    ,    13.108    ],
        7340    :[    68.759    ,    13.113    ],
        7350    :[    68.868    ,    13.117    ],
        7360    :[    68.977    ,    13.122    ],
        7370    :[    69.086    ,    13.126    ],
        7380    :[    69.194    ,    13.131    ],
        7390    :[    69.303    ,    13.136    ],
        7400    :[    69.412    ,    13.140    ],
        7410    :[    69.521    ,    13.145    ],
        7420    :[    69.630    ,    13.149    ],
        7430    :[    69.738    ,    13.154    ],
        7440    :[    69.847    ,    13.158    ],
        7450    :[    69.956    ,    13.163    ],
        7460    :[    70.065    ,    13.168    ],
        7470    :[    70.174    ,    13.172    ],
        7480    :[    70.282    ,    13.177    ],
        7490    :[    70.391    ,    13.181    ],
        7500    :[    70.500    ,    13.186    ],
        7510    :[    70.609    ,    13.191    ],
        7520    :[    70.718    ,    13.195    ],
        7530    :[    70.826    ,    13.200    ],
        7540    :[    70.935    ,    13.204    ],
        7550    :[    71.044    ,    13.209    ],
        7560    :[    71.153    ,    13.213    ],
        7570    :[    71.262    ,    13.218    ],
        7580    :[    71.370    ,    13.223    ],
        7590    :[    71.479    ,    13.227    ],
        7600    :[    71.588    ,    13.232    ],
        7610    :[    71.697    ,    13.236    ],
        7620    :[    71.806    ,    13.241    ],
        7630    :[    71.914    ,    13.245    ],
        7640    :[    72.023    ,    13.250    ],
        7650    :[    72.132    ,    13.255    ],
        7660    :[    72.241    ,    13.259    ],
        7670    :[    72.350    ,    13.264    ],
        7680    :[    72.458    ,    13.268    ],
        7690    :[    72.567    ,    13.273    ],
        7700    :[    72.676    ,    13.278    ],
        7710    :[    72.785    ,    13.282    ],
        7720    :[    72.894    ,    13.287    ],
        7730    :[    73.002    ,    13.291    ],
        7740    :[    73.111    ,    13.296    ],
        7750    :[    73.220    ,    13.300    ],
        7760    :[    73.329    ,    13.305    ],
        7770    :[    73.438    ,    13.310    ],
        7780    :[    73.546    ,    13.314    ],
        7790    :[    73.655    ,    13.319    ],
        7800    :[    73.764    ,    13.323    ],
        7810    :[    73.873    ,    13.328    ],
```

```
        7820    :[    73.982    ,    13.332    ],
        7830    :[    74.090    ,    13.337    ],
        7840    :[    74.199    ,    13.342    ],
        7850    :[    74.308    ,    13.346    ],
        7860    :[    74.417    ,    13.351    ],
        7870    :[    74.526    ,    13.355    ],
        7880    :[    74.634    ,    13.360    ],
        7890    :[    74.743    ,    13.364    ],
        7900    :[    74.852    ,    13.369    ],
        7910    :[    74.961    ,    13.374    ],
        7920    :[    75.070    ,    13.378    ],
        7930    :[    75.178    ,    13.383    ],
        7940    :[    75.287    ,    13.387    ],
        7950    :[    75.396    ,    13.392    ],
        7960    :[    75.505    ,    13.397    ],
        7970    :[    75.614    ,    13.401    ],
        7980    :[    75.722    ,    13.406    ],
        7990    :[    75.831    ,    13.410    ],
        8000    :[    75.940    ,    13.415    ],
        8010    :[    76.052    ,    13.420    ],
        8020    :[    76.164    ,    13.425    ],
        8030    :[    76.276    ,    13.430    ],
        8040    :[    76.388    ,    13.435    ],
        8050    :[    76.500    ,    13.440    ],
        8060    :[    76.612    ,    13.445    ],
        8070    :[    76.724    ,    13.450    ],
        8080    :[    76.836    ,    13.455    ],
        8090    :[    76.948    ,    13.460    ],
        8100    :[    77.060    ,    13.465    ],
        8110    :[    77.172    ,    13.470    ],
        8120    :[    77.284    ,    13.476    ],
        8130    :[    77.396    ,    13.481    ],
        8140    :[    77.508    ,    13.486    ],
        8150    :[    77.620    ,    13.491    ],
        8160    :[    77.732    ,    13.496    ],
        8170    :[    77.844    ,    13.501    ],
        8180    :[    77.956    ,    13.506    ],
        8190    :[    78.068    ,    13.511    ],
        8200    :[    78.180    ,    13.516    ],
        8210    :[    78.292    ,    13.521    ],
        8220    :[    78.404    ,    13.526    ],
        8230    :[    78.516    ,    13.531    ],
        8240    :[    78.628    ,    13.536    ],
        8250    :[    78.740    ,    13.541    ],
        8260    :[    78.852    ,    13.546    ],
        8270    :[    78.964    ,    13.551    ],
        8280    :[    79.076    ,    13.556    ],
        8290    :[    79.188    ,    13.562    ],
        8300    :[    79.300    ,    13.567    ],
        8310    :[    79.412    ,    13.572    ],
        8320    :[    79.524    ,    13.577    ],
        8330    :[    79.636    ,    13.582    ],
        8340    :[    79.748    ,    13.587    ],
        8350    :[    79.860    ,    13.592    ],
        8360    :[    79.972    ,    13.597    ],
        8370    :[    80.084    ,    13.602    ],
        8380    :[    80.196    ,    13.607    ],
        8390    :[    80.308    ,    13.612    ],
        8400    :[    80.420    ,    13.617    ],
        8410    :[    80.532    ,    13.622    ],
        8420    :[    80.644    ,    13.627    ],
        8430    :[    80.756    ,    13.632    ],
        8440    :[    80.868    ,    13.637    ],
        8450    :[    80.980    ,    13.642    ],
        8460    :[    81.092    ,    13.647    ],
        8470    :[    81.204    ,    13.653    ],
        8480    :[    81.316    ,    13.658    ],
        8490    :[    81.428    ,    13.663    ],
        8500    :[    81.540    ,    13.668    ],
        8510    :[    81.652    ,    13.673    ],
        8520    :[    81.764    ,    13.678    ],
        8530    :[    81.876    ,    13.683    ],
        8540    :[    81.988    ,    13.688    ],
        8550    :[    82.100    ,    13.693    ],
```

```
                                8560    :[    82.212    ,    13.698    ],
                                8570    :[    82.324    ,    13.703    ],
                                8580    :[    82.436    ,    13.708    ],
                                8590    :[    82.548    ,    13.713    ],
                                8600    :[    82.660    ,    13.718    ],
                                8610    :[    82.772    ,    13.723    ],
                                8620    :[    82.884    ,    13.728    ],
                                8630    :[    82.996    ,    13.733    ],
                                8640    :[    83.108    ,    13.738    ],
                                8650    :[    83.220    ,    13.744    ],
                                8660    :[    83.332    ,    13.749    ],
                                8670    :[    83.444    ,    13.754    ],
                                8680    :[    83.556    ,    13.759    ],
                                8690    :[    83.668    ,    13.764    ],
                                8700    :[    83.780    ,    13.769    ],
                                8710    :[    83.892    ,    13.774    ],
                                8720    :[    84.004    ,    13.779    ],
                                8730    :[    84.116    ,    13.784    ],
                                8740    :[    84.228    ,    13.789    ],
                                8750    :[    84.340    ,    13.794    ],
                                8760    :[    84.452    ,    13.799    ],
                                8770    :[    84.564    ,    13.804    ],
                                8780    :[    84.676    ,    13.809    ],
                                8790    :[    84.788    ,    13.814    ],
                                8800    :[    84.900    ,    13.819    ],
                                8810    :[    85.012    ,    13.824    ],
                                8820    :[    85.124    ,    13.830    ],
                                8830    :[    85.236    ,    13.835    ],
                                8840    :[    85.348    ,    13.840    ],
                                8850    :[    85.460    ,    13.845    ],
                                8860    :[    85.572    ,    13.850    ],
                                8870    :[    85.684    ,    13.855    ],
                                8880    :[    85.796    ,    13.860    ],
                                8890    :[    85.908    ,    13.865    ],
                                8900    :[    86.020    ,    13.870    ],
                                8910    :[    86.132    ,    13.875    ],
                                8920    :[    86.244    ,    13.880    ],
                                8930    :[    86.356    ,    13.885    ],
                                8940    :[    86.468    ,    13.890    ],
                                8950    :[    86.580    ,    13.895    ],
                                8960    :[    86.692    ,    13.900    ],
                                8970    :[    86.804    ,    13.905    ],
                                8980    :[    86.916    ,    13.910    ],
                                8990    :[    87.028    ,    13.915    ],
                                9000    :[    87.140    ,    13.921    ]}

    # Define density and viscosity for hydrogen, given pressure
    #                         Pressure    Density          Viscosity
    #                           (kPa)     (kg/m3)           (uPa-s)
    Hydrogen_properties =    {    10    :[    0.000032 ,    8.418    ],
                                  20    :[    0.00032  ,    8.418    ],
                                  30    :[    0.002    ,    8.418    ],
                                  40    :[    0.005    ,    8.418    ],
                                  50    :[    0.013    ,    8.418    ],
                                  60    :[    0.027    ,    8.418    ],
                                  70    :[    0.045    ,    8.418    ],
                                  80    :[    0.064    ,    8.418    ],
                                  90    :[    0.080    ,    8.418    ],
                                 100    :[    0.088    ,    8.418    ],
                                 110    :[    0.097    ,    8.418    ],
                                 120    :[    0.106    ,    8.418    ],
                                 130    :[    0.115    ,    8.418    ],
                                 140    :[    0.124    ,    8.419    ],
                                 150    :[    0.132    ,    8.419    ],
                                 160    :[    0.141    ,    8.419    ],
                                 170    :[    0.150    ,    8.419    ],
                                 180    :[    0.159    ,    8.420    ],
                                 190    :[    0.167    ,    8.420    ],
                                 200    :[    0.176    ,    8.420    ],
                                 210    :[    0.185    ,    8.421    ],
                                 220    :[    0.194    ,    8.421    ],
                                 230    :[    0.203    ,    8.421    ],
                                 240    :[    0.211    ,    8.421    ],
                                 250    :[    0.220    ,    8.422    ],
```

```
        260    :[    0.229    ,    8.422    ],
        270    :[    0.238    ,    8.422    ],
        280    :[    0.247    ,    8.422    ],
        290    :[    0.255    ,    8.423    ],
        300    :[    0.264    ,    8.423    ],
        310    :[    0.273    ,    8.423    ],
        320    :[    0.282    ,    8.423    ],
        330    :[    0.291    ,    8.424    ],
        340    :[    0.299    ,    8.424    ],
        350    :[    0.308    ,    8.424    ],
        360    :[    0.317    ,    8.424    ],
        370    :[    0.326    ,    8.425    ],
        380    :[    0.334    ,    8.425    ],
        390    :[    0.343    ,    8.425    ],
        400    :[    0.352    ,    8.425    ],
        410    :[    0.361    ,    8.426    ],
        420    :[    0.370    ,    8.426    ],
        430    :[    0.378    ,    8.426    ],
        440    :[    0.387    ,    8.427    ],
        450    :[    0.396    ,    8.427    ],
        460    :[    0.405    ,    8.427    ],
        470    :[    0.414    ,    8.427    ],
        480    :[    0.422    ,    8.428    ],
        490    :[    0.431    ,    8.428    ],
        500    :[    0.440    ,    8.428    ],
        510    :[    0.449    ,    8.428    ],
        520    :[    0.457    ,    8.429    ],
        530    :[    0.466    ,    8.429    ],
        540    :[    0.475    ,    8.429    ],
        550    :[    0.484    ,    8.429    ],
        560    :[    0.493    ,    8.430    ],
        570    :[    0.501    ,    8.430    ],
        580    :[    0.510    ,    8.430    ],
        590    :[    0.519    ,    8.430    ],
        600    :[    0.528    ,    8.431    ],
        610    :[    0.537    ,    8.431    ],
        620    :[    0.545    ,    8.431    ],
        630    :[    0.554    ,    8.431    ],
        640    :[    0.563    ,    8.432    ],
        650    :[    0.572    ,    8.432    ],
        660    :[    0.580    ,    8.432    ],
        670    :[    0.589    ,    8.433    ],
        680    :[    0.598    ,    8.433    ],
        690    :[    0.607    ,    8.433    ],
        700    :[    0.616    ,    8.433    ],
        710    :[    0.624    ,    8.434    ],
        720    :[    0.633    ,    8.434    ],
        730    :[    0.642    ,    8.434    ],
        740    :[    0.651    ,    8.434    ],
        750    :[    0.660    ,    8.435    ],
        760    :[    0.668    ,    8.435    ],
        770    :[    0.677    ,    8.435    ],
        780    :[    0.686    ,    8.435    ],
        790    :[    0.695    ,    8.436    ],
        800    :[    0.704    ,    8.436    ],
        810    :[    0.712    ,    8.436    ],
        820    :[    0.721    ,    8.436    ],
        830    :[    0.730    ,    8.437    ],
        840    :[    0.739    ,    8.437    ],
        850    :[    0.747    ,    8.437    ],
        860    :[    0.756    ,    8.437    ],
        870    :[    0.765    ,    8.438    ],
        880    :[    0.774    ,    8.438    ],
        890    :[    0.783    ,    8.438    ],
        900    :[    0.791    ,    8.439    ],
        910    :[    0.800    ,    8.439    ],
        920    :[    0.809    ,    8.439    ],
        930    :[    0.818    ,    8.439    ],
        940    :[    0.827    ,    8.440    ],
        950    :[    0.835    ,    8.440    ],
        960    :[    0.844    ,    8.440    ],
        970    :[    0.853    ,    8.440    ],
        980    :[    0.862    ,    8.441    ],
        990    :[    0.870    ,    8.441    ],
```

```
        1000    :[    0.879    ,    8.441    ],
        1010    :[    0.888    ,    8.441    ],
        1020    :[    0.896    ,    8.442    ],
        1030    :[    0.905    ,    8.442    ],
        1040    :[    0.913    ,    8.442    ],
        1050    :[    0.922    ,    8.442    ],
        1060    :[    0.930    ,    8.443    ],
        1070    :[    0.939    ,    8.443    ],
        1080    :[    0.947    ,    8.443    ],
        1090    :[    0.956    ,    8.443    ],
        1100    :[    0.965    ,    8.444    ],
        1110    :[    0.973    ,    8.444    ],
        1120    :[    0.982    ,    8.444    ],
        1130    :[    0.990    ,    8.445    ],
        1140    :[    0.999    ,    8.445    ],
        1150    :[    1.007    ,    8.445    ],
        1160    :[    1.016    ,    8.445    ],
        1170    :[    1.024    ,    8.446    ],
        1180    :[    1.033    ,    8.446    ],
        1190    :[    1.041    ,    8.446    ],
        1200    :[    1.050    ,    8.446    ],
        1210    :[    1.058    ,    8.447    ],
        1220    :[    1.067    ,    8.447    ],
        1230    :[    1.075    ,    8.447    ],
        1240    :[    1.084    ,    8.447    ],
        1250    :[    1.092    ,    8.448    ],
        1260    :[    1.101    ,    8.448    ],
        1270    :[    1.109    ,    8.448    ],
        1280    :[    1.118    ,    8.448    ],
        1290    :[    1.126    ,    8.449    ],
        1300    :[    1.135    ,    8.449    ],
        1310    :[    1.144    ,    8.449    ],
        1320    :[    1.152    ,    8.449    ],
        1330    :[    1.161    ,    8.450    ],
        1340    :[    1.169    ,    8.450    ],
        1350    :[    1.178    ,    8.450    ],
        1360    :[    1.186    ,    8.450    ],
        1370    :[    1.195    ,    8.451    ],
        1380    :[    1.203    ,    8.451    ],
        1390    :[    1.212    ,    8.451    ],
        1400    :[    1.220    ,    8.452    ],
        1410    :[    1.229    ,    8.452    ],
        1420    :[    1.237    ,    8.452    ],
        1430    :[    1.246    ,    8.452    ],
        1440    :[    1.254    ,    8.453    ],
        1450    :[    1.263    ,    8.453    ],
        1460    :[    1.271    ,    8.453    ],
        1470    :[    1.280    ,    8.453    ],
        1480    :[    1.288    ,    8.454    ],
        1490    :[    1.297    ,    8.454    ],
        1500    :[    1.306    ,    8.454    ],
        1510    :[    1.314    ,    8.454    ],
        1520    :[    1.323    ,    8.455    ],
        1530    :[    1.331    ,    8.455    ],
        1540    :[    1.340    ,    8.455    ],
        1550    :[    1.348    ,    8.455    ],
        1560    :[    1.357    ,    8.456    ],
        1570    :[    1.365    ,    8.456    ],
        1580    :[    1.374    ,    8.456    ],
        1590    :[    1.382    ,    8.456    ],
        1600    :[    1.391    ,    8.457    ],
        1610    :[    1.399    ,    8.457    ],
        1620    :[    1.408    ,    8.457    ],
        1630    :[    1.416    ,    8.457    ],
        1640    :[    1.425    ,    8.458    ],
        1650    :[    1.433    ,    8.458    ],
        1660    :[    1.442    ,    8.458    ],
        1670    :[    1.450    ,    8.459    ],
        1680    :[    1.459    ,    8.459    ],
        1690    :[    1.468    ,    8.459    ],
        1700    :[    1.476    ,    8.459    ],
        1710    :[    1.485    ,    8.460    ],
        1720    :[    1.493    ,    8.460    ],
        1730    :[    1.502    ,    8.460    ],
```

```
1740 :[ 1.510 , 8.460 ],
1750 :[ 1.519 , 8.461 ],
1760 :[ 1.527 , 8.461 ],
1770 :[ 1.536 , 8.461 ],
1780 :[ 1.544 , 8.461 ],
1790 :[ 1.553 , 8.462 ],
1800 :[ 1.561 , 8.462 ],
1810 :[ 1.570 , 8.462 ],
1820 :[ 1.578 , 8.462 ],
1830 :[ 1.587 , 8.463 ],
1840 :[ 1.595 , 8.463 ],
1850 :[ 1.604 , 8.463 ],
1860 :[ 1.612 , 8.463 ],
1870 :[ 1.621 , 8.464 ],
1880 :[ 1.629 , 8.464 ],
1890 :[ 1.638 , 8.464 ],
1900 :[ 1.647 , 8.464 ],
1910 :[ 1.655 , 8.465 ],
1920 :[ 1.664 , 8.465 ],
1930 :[ 1.672 , 8.465 ],
1940 :[ 1.681 , 8.466 ],
1950 :[ 1.689 , 8.466 ],
1960 :[ 1.698 , 8.466 ],
1970 :[ 1.706 , 8.466 ],
1980 :[ 1.715 , 8.467 ],
1990 :[ 1.723 , 8.467 ],
2000 :[ 1.732 , 8.467 ],
2010 :[ 1.740 , 8.467 ],
2020 :[ 1.749 , 8.468 ],
2030 :[ 1.757 , 8.468 ],
2040 :[ 1.766 , 8.468 ],
2050 :[ 1.774 , 8.468 ],
2060 :[ 1.783 , 8.469 ],
2070 :[ 1.791 , 8.469 ],
2080 :[ 1.800 , 8.469 ],
2090 :[ 1.809 , 8.469 ],
2100 :[ 1.817 , 8.470 ],
2110 :[ 1.826 , 8.470 ],
2120 :[ 1.834 , 8.470 ],
2130 :[ 1.843 , 8.470 ],
2140 :[ 1.851 , 8.471 ],
2150 :[ 1.860 , 8.471 ],
2160 :[ 1.868 , 8.471 ],
2170 :[ 1.877 , 8.471 ],
2180 :[ 1.885 , 8.472 ],
2190 :[ 1.894 , 8.472 ],
2200 :[ 1.902 , 8.472 ],
2210 :[ 1.911 , 8.473 ],
2220 :[ 1.919 , 8.473 ],
2230 :[ 1.928 , 8.473 ],
2240 :[ 1.936 , 8.473 ],
2250 :[ 1.945 , 8.474 ],
2260 :[ 1.953 , 8.474 ],
2270 :[ 1.962 , 8.474 ],
2280 :[ 1.971 , 8.474 ],
2290 :[ 1.979 , 8.475 ],
2300 :[ 1.988 , 8.475 ],
2310 :[ 1.996 , 8.475 ],
2320 :[ 2.005 , 8.475 ],
2330 :[ 2.013 , 8.476 ],
2340 :[ 2.022 , 8.476 ],
2350 :[ 2.030 , 8.476 ],
2360 :[ 2.039 , 8.476 ],
2370 :[ 2.047 , 8.477 ],
2380 :[ 2.056 , 8.477 ],
2390 :[ 2.064 , 8.477 ],
2400 :[ 2.073 , 8.477 ],
2410 :[ 2.081 , 8.478 ],
2420 :[ 2.090 , 8.478 ],
2430 :[ 2.098 , 8.478 ],
2440 :[ 2.107 , 8.478 ],
2450 :[ 2.115 , 8.479 ],
2460 :[ 2.124 , 8.479 ],
2470 :[ 2.132 , 8.479 ],
```

```
        2480    :[    2.141    ,    8.480    ],
        2490    :[    2.150    ,    8.480    ],
        2500    :[    2.158    ,    8.480    ],
        2510    :[    2.167    ,    8.480    ],
        2520    :[    2.175    ,    8.481    ],
        2530    :[    2.184    ,    8.481    ],
        2540    :[    2.192    ,    8.481    ],
        2550    :[    2.201    ,    8.481    ],
        2560    :[    2.209    ,    8.482    ],
        2570    :[    2.218    ,    8.482    ],
        2580    :[    2.226    ,    8.482    ],
        2590    :[    2.235    ,    8.482    ],
        2600    :[    2.243    ,    8.483    ],
        2610    :[    2.252    ,    8.483    ],
        2620    :[    2.260    ,    8.483    ],
        2630    :[    2.269    ,    8.483    ],
        2640    :[    2.277    ,    8.484    ],
        2650    :[    2.286    ,    8.484    ],
        2660    :[    2.294    ,    8.484    ],
        2670    :[    2.303    ,    8.484    ],
        2680    :[    2.312    ,    8.485    ],
        2690    :[    2.320    ,    8.485    ],
        2700    :[    2.329    ,    8.485    ],
        2710    :[    2.337    ,    8.485    ],
        2720    :[    2.346    ,    8.486    ],
        2730    :[    2.354    ,    8.486    ],
        2740    :[    2.363    ,    8.486    ],
        2750    :[    2.371    ,    8.487    ],
        2760    :[    2.380    ,    8.487    ],
        2770    :[    2.388    ,    8.487    ],
        2780    :[    2.397    ,    8.487    ],
        2790    :[    2.405    ,    8.488    ],
        2800    :[    2.414    ,    8.488    ],
        2810    :[    2.422    ,    8.488    ],
        2820    :[    2.431    ,    8.488    ],
        2830    :[    2.439    ,    8.489    ],
        2840    :[    2.448    ,    8.489    ],
        2850    :[    2.456    ,    8.489    ],
        2860    :[    2.465    ,    8.489    ],
        2870    :[    2.474    ,    8.490    ],
        2880    :[    2.482    ,    8.490    ],
        2890    :[    2.491    ,    8.490    ],
        2900    :[    2.499    ,    8.490    ],
        2910    :[    2.508    ,    8.491    ],
        2920    :[    2.516    ,    8.491    ],
        2930    :[    2.525    ,    8.491    ],
        2940    :[    2.533    ,    8.491    ],
        2950    :[    2.542    ,    8.492    ],
        2960    :[    2.550    ,    8.492    ],
        2970    :[    2.559    ,    8.492    ],
        2980    :[    2.567    ,    8.492    ],
        2990    :[    2.576    ,    8.493    ],
        3000    :[    2.584    ,    8.493    ],
        3010    :[    2.593    ,    8.493    ],
        3020    :[    2.601    ,    8.493    ],
        3030    :[    2.610    ,    8.494    ],
        3040    :[    2.618    ,    8.494    ],
        3050    :[    2.627    ,    8.494    ],
        3060    :[    2.635    ,    8.495    ],
        3070    :[    2.644    ,    8.495    ],
        3080    :[    2.653    ,    8.495    ],
        3090    :[    2.661    ,    8.495    ],
        3100    :[    2.670    ,    8.496    ],
        3110    :[    2.678    ,    8.496    ],
        3120    :[    2.687    ,    8.496    ],
        3130    :[    2.695    ,    8.496    ],
        3140    :[    2.704    ,    8.497    ],
        3150    :[    2.712    ,    8.497    ],
        3160    :[    2.721    ,    8.497    ],
        3170    :[    2.729    ,    8.497    ],
        3180    :[    2.738    ,    8.498    ],
        3190    :[    2.746    ,    8.498    ],
        3200    :[    2.755    ,    8.498    ],
        3210    :[    2.763    ,    8.498    ],
```

```
        3220    :[    2.772    ,    8.499    ],
        3230    :[    2.780    ,    8.499    ],
        3240    :[    2.789    ,    8.499    ],
        3250    :[    2.797    ,    8.499    ],
        3260    :[    2.806    ,    8.500    ],
        3270    :[    2.815    ,    8.500    ],
        3280    :[    2.823    ,    8.500    ],
        3290    :[    2.832    ,    8.500    ],
        3300    :[    2.840    ,    8.501    ],
        3310    :[    2.849    ,    8.501    ],
        3320    :[    2.857    ,    8.501    ],
        3330    :[    2.866    ,    8.502    ],
        3340    :[    2.874    ,    8.502    ],
        3350    :[    2.883    ,    8.502    ],
        3360    :[    2.891    ,    8.502    ],
        3370    :[    2.900    ,    8.503    ],
        3380    :[    2.908    ,    8.503    ],
        3390    :[    2.917    ,    8.503    ],
        3400    :[    2.925    ,    8.503    ],
        3410    :[    2.934    ,    8.504    ],
        3420    :[    2.942    ,    8.504    ],
        3430    :[    2.951    ,    8.504    ],
        3440    :[    2.959    ,    8.504    ],
        3450    :[    2.968    ,    8.505    ],
        3460    :[    2.977    ,    8.505    ],
        3470    :[    2.985    ,    8.505    ],
        3480    :[    2.994    ,    8.505    ],
        3490    :[    3.002    ,    8.506    ],
        3500    :[    3.011    ,    8.506    ],
        3510    :[    3.019    ,    8.506    ],
        3520    :[    3.028    ,    8.506    ],
        3530    :[    3.036    ,    8.507    ],
        3540    :[    3.045    ,    8.507    ],
        3550    :[    3.053    ,    8.507    ],
        3560    :[    3.062    ,    8.507    ],
        3570    :[    3.070    ,    8.508    ],
        3580    :[    3.079    ,    8.508    ],
        3590    :[    3.087    ,    8.508    ],
        3600    :[    3.096    ,    8.509    ],
        3610    :[    3.104    ,    8.509    ],
        3620    :[    3.113    ,    8.509    ],
        3630    :[    3.121    ,    8.509    ],
        3640    :[    3.130    ,    8.510    ],
        3650    :[    3.138    ,    8.510    ],
        3660    :[    3.147    ,    8.510    ],
        3670    :[    3.156    ,    8.510    ],
        3680    :[    3.164    ,    8.511    ],
        3690    :[    3.173    ,    8.511    ],
        3700    :[    3.181    ,    8.511    ],
        3710    :[    3.190    ,    8.511    ],
        3720    :[    3.198    ,    8.512    ],
        3730    :[    3.207    ,    8.512    ],
        3740    :[    3.215    ,    8.512    ],
        3750    :[    3.224    ,    8.512    ],
        3760    :[    3.232    ,    8.513    ],
        3770    :[    3.241    ,    8.513    ],
        3780    :[    3.249    ,    8.513    ],
        3790    :[    3.258    ,    8.513    ],
        3800    :[    3.266    ,    8.514    ],
        3810    :[    3.275    ,    8.514    ],
        3820    :[    3.283    ,    8.514    ],
        3830    :[    3.292    ,    8.514    ],
        3840    :[    3.300    ,    8.515    ],
        3850    :[    3.309    ,    8.515    ],
        3860    :[    3.318    ,    8.515    ],
        3870    :[    3.326    ,    8.516    ],
        3880    :[    3.335    ,    8.516    ],
        3890    :[    3.343    ,    8.516    ],
        3900    :[    3.352    ,    8.516    ],
        3910    :[    3.360    ,    8.517    ],
        3920    :[    3.369    ,    8.517    ],
        3930    :[    3.377    ,    8.517    ],
        3940    :[    3.386    ,    8.517    ],
        3950    :[    3.394    ,    8.518    ],
```

```
        3960    :[    3.403    ,    8.518    ],
        3970    :[    3.411    ,    8.518    ],
        3980    :[    3.420    ,    8.518    ],
        3990    :[    3.428    ,    8.519    ],
        4000    :[    3.437    ,    8.519    ],
        4010    :[    3.445    ,    8.519    ],
        4020    :[    3.454    ,    8.519    ],
        4030    :[    3.462    ,    8.520    ],
        4040    :[    3.471    ,    8.520    ],
        4050    :[    3.480    ,    8.520    ],
        4060    :[    3.488    ,    8.520    ],
        4070    :[    3.497    ,    8.521    ],
        4080    :[    3.505    ,    8.521    ],
        4090    :[    3.514    ,    8.521    ],
        4100    :[    3.522    ,    8.521    ],
        4110    :[    3.531    ,    8.522    ],
        4120    :[    3.539    ,    8.522    ],
        4130    :[    3.548    ,    8.522    ],
        4140    :[    3.556    ,    8.523    ],
        4150    :[    3.565    ,    8.523    ],
        4160    :[    3.573    ,    8.523    ],
        4170    :[    3.582    ,    8.523    ],
        4180    :[    3.590    ,    8.524    ],
        4190    :[    3.599    ,    8.524    ],
        4200    :[    3.607    ,    8.524    ],
        4210    :[    3.616    ,    8.524    ],
        4220    :[    3.624    ,    8.525    ],
        4230    :[    3.633    ,    8.525    ],
        4240    :[    3.641    ,    8.525    ],
        4250    :[    3.650    ,    8.525    ],
        4260    :[    3.659    ,    8.526    ],
        4270    :[    3.667    ,    8.526    ],
        4280    :[    3.676    ,    8.526    ],
        4290    :[    3.684    ,    8.526    ],
        4300    :[    3.693    ,    8.527    ],
        4310    :[    3.701    ,    8.527    ],
        4320    :[    3.710    ,    8.527    ],
        4330    :[    3.718    ,    8.527    ],
        4340    :[    3.727    ,    8.528    ],
        4350    :[    3.735    ,    8.528    ],
        4360    :[    3.744    ,    8.528    ],
        4370    :[    3.752    ,    8.528    ],
        4380    :[    3.761    ,    8.529    ],
        4390    :[    3.769    ,    8.529    ],
        4400    :[    3.778    ,    8.529    ],
        4410    :[    3.786    ,    8.530    ],
        4420    :[    3.795    ,    8.530    ],
        4430    :[    3.803    ,    8.530    ],
        4440    :[    3.812    ,    8.530    ],
        4450    :[    3.821    ,    8.531    ],
        4460    :[    3.829    ,    8.531    ],
        4470    :[    3.838    ,    8.531    ],
        4480    :[    3.846    ,    8.531    ],
        4490    :[    3.855    ,    8.532    ],
        4500    :[    3.863    ,    8.532    ],
        4510    :[    3.872    ,    8.532    ],
        4520    :[    3.880    ,    8.532    ],
        4530    :[    3.889    ,    8.533    ],
        4540    :[    3.897    ,    8.533    ],
        4550    :[    3.906    ,    8.533    ],
        4560    :[    3.914    ,    8.533    ],
        4570    :[    3.923    ,    8.534    ],
        4580    :[    3.931    ,    8.534    ],
        4590    :[    3.940    ,    8.534    ],
        4600    :[    3.948    ,    8.534    ],
        4610    :[    3.957    ,    8.535    ],
        4620    :[    3.965    ,    8.535    ],
        4630    :[    3.974    ,    8.535    ],
        4640    :[    3.983    ,    8.535    ],
        4650    :[    3.991    ,    8.536    ],
        4660    :[    4.000    ,    8.536    ],
        4670    :[    4.008    ,    8.536    ],
        4680    :[    4.017    ,    8.537    ],
        4690    :[    4.025    ,    8.537    ],
```

```
            4700    :[    4.034    ,    8.537    ],
            4710    :[    4.042    ,    8.537    ],
            4720    :[    4.051    ,    8.538    ],
            4730    :[    4.059    ,    8.538    ],
            4740    :[    4.068    ,    8.538    ],
            4750    :[    4.076    ,    8.538    ],
            4760    :[    4.085    ,    8.539    ],
            4770    :[    4.093    ,    8.539    ],
            4780    :[    4.102    ,    8.539    ],
            4790    :[    4.110    ,    8.539    ],
            4800    :[    4.119    ,    8.540    ],
            4810    :[    4.127    ,    8.540    ],
            4820    :[    4.136    ,    8.540    ],
            4830    :[    4.144    ,    8.540    ],
            4840    :[    4.153    ,    8.541    ],
            4850    :[    4.162    ,    8.541    ],
            4860    :[    4.170    ,    8.541    ],
            4870    :[    4.179    ,    8.541    ],
            4880    :[    4.187    ,    8.542    ],
            4890    :[    4.196    ,    8.542    ],
            4900    :[    4.204    ,    8.542    ],
            4910    :[    4.213    ,    8.542    ],
            4920    :[    4.221    ,    8.543    ],
            4930    :[    4.230    ,    8.543    ],
            4940    :[    4.238    ,    8.543    ],
            4950    :[    4.247    ,    8.544    ],
            4960    :[    4.255    ,    8.544    ],
            4970    :[    4.264    ,    8.544    ],
            4980    :[    4.272    ,    8.544    ],
            4990    :[    4.281    ,    8.545    ],
            5000    :[    4.289    ,    8.545    ],
            5010    :[    4.297    ,    8.545    ],
            5020    :[    4.306    ,    8.545    ],
            5030    :[    4.314    ,    8.546    ],
            5040    :[    4.322    ,    8.546    ],
            5050    :[    4.330    ,    8.546    ],
            5060    :[    4.338    ,    8.546    ],
            5070    :[    4.346    ,    8.547    ],
            5080    :[    4.354    ,    8.547    ],
            5090    :[    4.362    ,    8.547    ],
            5100    :[    4.370    ,    8.547    ],
            5110    :[    4.378    ,    8.548    ],
            5120    :[    4.386    ,    8.548    ],
            5130    :[    4.394    ,    8.548    ],
            5140    :[    4.402    ,    8.549    ],
            5150    :[    4.410    ,    8.549    ],
            5160    :[    4.418    ,    8.549    ],
            5170    :[    4.426    ,    8.549    ],
            5180    :[    4.434    ,    8.550    ],
            5190    :[    4.442    ,    8.550    ],
            5200    :[    4.451    ,    8.550    ],
            5210    :[    4.459    ,    8.550    ],
            5220    :[    4.467    ,    8.551    ],
            5230    :[    4.475    ,    8.551    ],
            5240    :[    4.483    ,    8.551    ],
            5250    :[    4.491    ,    8.551    ],
            5260    :[    4.499    ,    8.552    ],
            5270    :[    4.507    ,    8.552    ],
            5280    :[    4.515    ,    8.552    ],
            5290    :[    4.523    ,    8.553    ],
            5300    :[    4.531    ,    8.553    ],
            5310    :[    4.539    ,    8.553    ],
            5320    :[    4.547    ,    8.553    ],
            5330    :[    4.555    ,    8.554    ],
            5340    :[    4.563    ,    8.554    ],
            5350    :[    4.571    ,    8.554    ],
            5360    :[    4.579    ,    8.554    ],
            5370    :[    4.588    ,    8.555    ],
            5380    :[    4.596    ,    8.555    ],
            5390    :[    4.604    ,    8.555    ],
            5400    :[    4.612    ,    8.555    ],
            5410    :[    4.620    ,    8.556    ],
            5420    :[    4.628    ,    8.556    ],
            5430    :[    4.636    ,    8.556    ],
```

```
                5440    :[    4.644    ,    8.557    ],
                5450    :[    4.652    ,    8.557    ],
                5460    :[    4.660    ,    8.557    ],
                5470    :[    4.668    ,    8.557    ],
                5480    :[    4.676    ,    8.558    ],
                5490    :[    4.684    ,    8.558    ],
                5500    :[    4.692    ,    8.558    ],
                5510    :[    4.700    ,    8.558    ],
                5520    :[    4.708    ,    8.559    ],
                5530    :[    4.716    ,    8.559    ],
                5540    :[    4.724    ,    8.559    ],
                5550    :[    4.733    ,    8.559    ],
                5560    :[    4.741    ,    8.560    ],
                5570    :[    4.749    ,    8.560    ],
                5580    :[    4.757    ,    8.560    ],
                5590    :[    4.765    ,    8.561    ],
                5600    :[    4.773    ,    8.561    ],
                5610    :[    4.781    ,    8.561    ],
                5620    :[    4.789    ,    8.561    ],
                5630    :[    4.797    ,    8.562    ],
                5640    :[    4.805    ,    8.562    ],
                5650    :[    4.813    ,    8.562    ],
                5660    :[    4.821    ,    8.562    ],
                5670    :[    4.829    ,    8.563    ],
                5680    :[    4.837    ,    8.563    ],
                5690    :[    4.845    ,    8.563    ],
                5700    :[    4.853    ,    8.563    ],
                5710    :[    4.861    ,    8.564    ],
                5720    :[    4.869    ,    8.564    ],
                5730    :[    4.878    ,    8.564    ],
                5740    :[    4.886    ,    8.565    ],
                5750    :[    4.894    ,    8.565    ],
                5760    :[    4.902    ,    8.565    ],
                5770    :[    4.910    ,    8.565    ],
                5780    :[    4.918    ,    8.566    ],
                5790    :[    4.926    ,    8.566    ],
                5800    :[    4.934    ,    8.566    ],
                5810    :[    4.942    ,    8.566    ],
                5820    :[    4.950    ,    8.567    ],
                5830    :[    4.958    ,    8.567    ],
                5840    :[    4.966    ,    8.567    ],
                5850    :[    4.974    ,    8.567    ],
                5860    :[    4.982    ,    8.568    ],
                5870    :[    4.990    ,    8.568    ],
                5880    :[    4.998    ,    8.568    ],
                5890    :[    5.006    ,    8.568    ],
                5900    :[    5.015    ,    8.569    ],
                5910    :[    5.023    ,    8.569    ],
                5920    :[    5.031    ,    8.569    ],
                5930    :[    5.039    ,    8.570    ],
                5940    :[    5.047    ,    8.570    ],
                5950    :[    5.055    ,    8.570    ],
                5960    :[    5.063    ,    8.570    ],
                5970    :[    5.071    ,    8.571    ],
                5980    :[    5.079    ,    8.571    ],
                5990    :[    5.087    ,    8.571    ],
                6000    :[    5.095    ,    8.571    ],
                6010    :[    5.103    ,    8.572    ],
                6020    :[    5.111    ,    8.572    ],
                6030    :[    5.119    ,    8.572    ],
                6040    :[    5.127    ,    8.572    ],
                6050    :[    5.135    ,    8.573    ],
                6060    :[    5.143    ,    8.573    ],
                6070    :[    5.151    ,    8.573    ],
                6080    :[    5.160    ,    8.574    ],
                6090    :[    5.168    ,    8.574    ],
                6100    :[    5.176    ,    8.574    ],
                6110    :[    5.184    ,    8.574    ],
                6120    :[    5.192    ,    8.575    ],
                6130    :[    5.200    ,    8.575    ],
                6140    :[    5.208    ,    8.575    ],
                6150    :[    5.216    ,    8.575    ],
                6160    :[    5.224    ,    8.576    ],
                6170    :[    5.232    ,    8.576    ],
```

```
        6180    :[    5.240    ,    8.576    ],
        6190    :[    5.248    ,    8.576    ],
        6200    :[    5.256    ,    8.577    ],
        6210    :[    5.264    ,    8.577    ],
        6220    :[    5.272    ,    8.577    ],
        6230    :[    5.280    ,    8.578    ],
        6240    :[    5.288    ,    8.578    ],
        6250    :[    5.297    ,    8.578    ],
        6260    :[    5.305    ,    8.578    ],
        6270    :[    5.313    ,    8.579    ],
        6280    :[    5.321    ,    8.579    ],
        6290    :[    5.329    ,    8.579    ],
        6300    :[    5.337    ,    8.579    ],
        6310    :[    5.345    ,    8.580    ],
        6320    :[    5.353    ,    8.580    ],
        6330    :[    5.361    ,    8.580    ],
        6340    :[    5.369    ,    8.580    ],
        6350    :[    5.377    ,    8.581    ],
        6360    :[    5.385    ,    8.581    ],
        6370    :[    5.393    ,    8.581    ],
        6380    :[    5.401    ,    8.582    ],
        6390    :[    5.409    ,    8.582    ],
        6400    :[    5.417    ,    8.582    ],
        6410    :[    5.425    ,    8.582    ],
        6420    :[    5.433    ,    8.583    ],
        6430    :[    5.442    ,    8.583    ],
        6440    :[    5.450    ,    8.583    ],
        6450    :[    5.458    ,    8.583    ],
        6460    :[    5.466    ,    8.584    ],
        6470    :[    5.474    ,    8.584    ],
        6480    :[    5.482    ,    8.584    ],
        6490    :[    5.490    ,    8.584    ],
        6500    :[    5.498    ,    8.585    ],
        6510    :[    5.506    ,    8.585    ],
        6520    :[    5.514    ,    8.585    ],
        6530    :[    5.522    ,    8.586    ],
        6540    :[    5.530    ,    8.586    ],
        6550    :[    5.538    ,    8.586    ],
        6560    :[    5.546    ,    8.586    ],
        6570    :[    5.554    ,    8.587    ],
        6580    :[    5.562    ,    8.587    ],
        6590    :[    5.570    ,    8.587    ],
        6600    :[    5.578    ,    8.587    ],
        6610    :[    5.587    ,    8.588    ],
        6620    :[    5.595    ,    8.588    ],
        6630    :[    5.603    ,    8.588    ],
        6640    :[    5.611    ,    8.588    ],
        6650    :[    5.619    ,    8.589    ],
        6660    :[    5.627    ,    8.589    ],
        6670    :[    5.635    ,    8.589    ],
        6680    :[    5.643    ,    8.590    ],
        6690    :[    5.651    ,    8.590    ],
        6700    :[    5.659    ,    8.590    ],
        6710    :[    5.667    ,    8.590    ],
        6720    :[    5.675    ,    8.591    ],
        6730    :[    5.683    ,    8.591    ],
        6740    :[    5.691    ,    8.591    ],
        6750    :[    5.699    ,    8.591    ],
        6760    :[    5.707    ,    8.592    ],
        6770    :[    5.715    ,    8.592    ],
        6780    :[    5.724    ,    8.592    ],
        6790    :[    5.732    ,    8.592    ],
        6800    :[    5.740    ,    8.593    ],
        6810    :[    5.748    ,    8.593    ],
        6820    :[    5.756    ,    8.593    ],
        6830    :[    5.764    ,    8.594    ],
        6840    :[    5.772    ,    8.594    ],
        6850    :[    5.780    ,    8.594    ],
        6860    :[    5.788    ,    8.594    ],
        6870    :[    5.796    ,    8.595    ],
        6880    :[    5.804    ,    8.595    ],
        6890    :[    5.812    ,    8.595    ],
        6900    :[    5.820    ,    8.595    ],
        6910    :[    5.828    ,    8.596    ],
```

```
6920 :[ 5.836 , 8.596 ],
6930 :[ 5.844 , 8.596 ],
6940 :[ 5.852 , 8.596 ],
6950 :[ 5.860 , 8.597 ],
6960 :[ 5.869 , 8.597 ],
6970 :[ 5.877 , 8.597 ],
6980 :[ 5.885 , 8.598 ],
6990 :[ 5.893 , 8.598 ],
7000 :[ 5.901 , 8.598 ],
7010 :[ 5.909 , 8.598 ],
7020 :[ 5.917 , 8.599 ],
7030 :[ 5.925 , 8.599 ],
7040 :[ 5.933 , 8.599 ],
7050 :[ 5.941 , 8.599 ],
7060 :[ 5.949 , 8.600 ],
7070 :[ 5.957 , 8.600 ],
7080 :[ 5.965 , 8.600 ],
7090 :[ 5.973 , 8.600 ],
7100 :[ 5.981 , 8.601 ],
7110 :[ 5.989 , 8.601 ],
7120 :[ 5.997 , 8.601 ],
7130 :[ 6.005 , 8.602 ],
7140 :[ 6.014 , 8.602 ],
7150 :[ 6.022 , 8.602 ],
7160 :[ 6.030 , 8.602 ],
7170 :[ 6.038 , 8.603 ],
7180 :[ 6.046 , 8.603 ],
7190 :[ 6.054 , 8.603 ],
7200 :[ 6.062 , 8.603 ],
7210 :[ 6.070 , 8.604 ],
7220 :[ 6.078 , 8.604 ],
7230 :[ 6.086 , 8.604 ],
7240 :[ 6.094 , 8.604 ],
7250 :[ 6.102 , 8.605 ],
7260 :[ 6.110 , 8.605 ],
7270 :[ 6.118 , 8.605 ],
7280 :[ 6.126 , 8.605 ],
7290 :[ 6.134 , 8.606 ],
7300 :[ 6.142 , 8.606 ],
7310 :[ 6.151 , 8.606 ],
7320 :[ 6.159 , 8.607 ],
7330 :[ 6.167 , 8.607 ],
7340 :[ 6.175 , 8.607 ],
7350 :[ 6.183 , 8.607 ],
7360 :[ 6.191 , 8.608 ],
7370 :[ 6.199 , 8.608 ],
7380 :[ 6.207 , 8.608 ],
7390 :[ 6.215 , 8.608 ],
7400 :[ 6.223 , 8.609 ],
7410 :[ 6.231 , 8.609 ],
7420 :[ 6.239 , 8.609 ],
7430 :[ 6.247 , 8.609 ],
7440 :[ 6.255 , 8.610 ],
7450 :[ 6.263 , 8.610 ],
7460 :[ 6.271 , 8.610 ],
7470 :[ 6.279 , 8.611 ],
7480 :[ 6.287 , 8.611 ],
7490 :[ 6.296 , 8.611 ],
7500 :[ 6.304 , 8.611 ],
7510 :[ 6.312 , 8.612 ],
7520 :[ 6.320 , 8.612 ],
7530 :[ 6.328 , 8.612 ],
7540 :[ 6.336 , 8.612 ],
7550 :[ 6.344 , 8.613 ],
7560 :[ 6.352 , 8.613 ],
7570 :[ 6.360 , 8.613 ],
7580 :[ 6.368 , 8.613 ],
7590 :[ 6.376 , 8.614 ],
7600 :[ 6.384 , 8.614 ],
7610 :[ 6.392 , 8.614 ],
7620 :[ 6.400 , 8.615 ],
7630 :[ 6.408 , 8.615 ],
7640 :[ 6.416 , 8.615 ],
7650 :[ 6.424 , 8.615 ],
```

```
        7660    :[    6.432    ,    8.616    ],
        7670    :[    6.441    ,    8.616    ],
        7680    :[    6.449    ,    8.616    ],
        7690    :[    6.457    ,    8.616    ],
        7700    :[    6.465    ,    8.617    ],
        7710    :[    6.473    ,    8.617    ],
        7720    :[    6.481    ,    8.617    ],
        7730    :[    6.489    ,    8.617    ],
        7740    :[    6.497    ,    8.618    ],
        7750    :[    6.505    ,    8.618    ],
        7760    :[    6.513    ,    8.618    ],
        7770    :[    6.521    ,    8.619    ],
        7780    :[    6.529    ,    8.619    ],
        7790    :[    6.537    ,    8.619    ],
        7800    :[    6.545    ,    8.619    ],
        7810    :[    6.553    ,    8.620    ],
        7820    :[    6.561    ,    8.620    ],
        7830    :[    6.569    ,    8.620    ],
        7840    :[    6.578    ,    8.620    ],
        7850    :[    6.586    ,    8.621    ],
        7860    :[    6.594    ,    8.621    ],
        7870    :[    6.602    ,    8.621    ],
        7880    :[    6.610    ,    8.621    ],
        7890    :[    6.618    ,    8.622    ],
        7900    :[    6.626    ,    8.622    ],
        7910    :[    6.634    ,    8.622    ],
        7920    :[    6.642    ,    8.623    ],
        7930    :[    6.650    ,    8.623    ],
        7940    :[    6.658    ,    8.623    ],
        7950    :[    6.666    ,    8.623    ],
        7960    :[    6.674    ,    8.624    ],
        7970    :[    6.682    ,    8.624    ],
        7980    :[    6.690    ,    8.624    ],
        7990    :[    6.698    ,    8.624    ],
        8000    :[    6.706    ,    8.625    ],
        8010    :[    6.714    ,    8.625    ],
        8020    :[    6.723    ,    8.625    ],
        8030    :[    6.731    ,    8.625    ],
        8040    :[    6.739    ,    8.626    ],
        8050    :[    6.747    ,    8.626    ],
        8060    :[    6.755    ,    8.626    ],
        8070    :[    6.763    ,    8.627    ],
        8080    :[    6.771    ,    8.627    ],
        8090    :[    6.779    ,    8.627    ],
        8100    :[    6.787    ,    8.627    ],
        8110    :[    6.795    ,    8.628    ],
        8120    :[    6.803    ,    8.628    ],
        8130    :[    6.811    ,    8.628    ],
        8140    :[    6.819    ,    8.628    ],
        8150    :[    6.827    ,    8.629    ],
        8160    :[    6.835    ,    8.629    ],
        8170    :[    6.843    ,    8.629    ],
        8180    :[    6.851    ,    8.629    ],
        8190    :[    6.859    ,    8.630    ],
        8200    :[    6.868    ,    8.630    ],
        8210    :[    6.876    ,    8.630    ],
        8220    :[    6.884    ,    8.631    ],
        8230    :[    6.892    ,    8.631    ],
        8240    :[    6.900    ,    8.631    ],
        8250    :[    6.908    ,    8.631    ],
        8260    :[    6.916    ,    8.632    ],
        8270    :[    6.924    ,    8.632    ],
        8280    :[    6.932    ,    8.632    ],
        8290    :[    6.940    ,    8.632    ],
        8300    :[    6.948    ,    8.633    ],
        8310    :[    6.956    ,    8.633    ],
        8320    :[    6.964    ,    8.633    ],
        8330    :[    6.972    ,    8.633    ],
        8340    :[    6.980    ,    8.634    ],
        8350    :[    6.988    ,    8.634    ],
        8360    :[    6.996    ,    8.634    ],
        8370    :[    7.005    ,    8.635    ],
        8380    :[    7.013    ,    8.635    ],
        8390    :[    7.021    ,    8.635    ],
```

```
                                8400    :[    7.029    ,    8.635    ],
                                8410    :[    7.037    ,    8.636    ],
                                8420    :[    7.045    ,    8.636    ],
                                8430    :[    7.053    ,    8.636    ],
                                8440    :[    7.061    ,    8.636    ],
                                8450    :[    7.069    ,    8.637    ],
                                8460    :[    7.077    ,    8.637    ],
                                8470    :[    7.085    ,    8.637    ],
                                8480    :[    7.093    ,    8.637    ],
                                8490    :[    7.101    ,    8.638    ],
                                8500    :[    7.109    ,    8.638    ],
                                8510    :[    7.117    ,    8.638    ],
                                8520    :[    7.125    ,    8.639    ],
                                8530    :[    7.133    ,    8.639    ],
                                8540    :[    7.141    ,    8.639    ],
                                8550    :[    7.150    ,    8.639    ],
                                8560    :[    7.158    ,    8.640    ],
                                8570    :[    7.166    ,    8.640    ],
                                8580    :[    7.174    ,    8.640    ],
                                8590    :[    7.182    ,    8.640    ],
                                8600    :[    7.190    ,    8.641    ],
                                8610    :[    7.198    ,    8.641    ],
                                8620    :[    7.206    ,    8.641    ],
                                8630    :[    7.214    ,    8.641    ],
                                8640    :[    7.222    ,    8.642    ],
                                8650    :[    7.230    ,    8.642    ],
                                8660    :[    7.238    ,    8.642    ],
                                8670    :[    7.246    ,    8.643    ],
                                8680    :[    7.254    ,    8.643    ],
                                8690    :[    7.262    ,    8.643    ],
                                8700    :[    7.270    ,    8.643    ],
                                8710    :[    7.278    ,    8.644    ],
                                8720    :[    7.287    ,    8.644    ],
                                8730    :[    7.295    ,    8.644    ],
                                8740    :[    7.303    ,    8.644    ],
                                8750    :[    7.311    ,    8.645    ],
                                8760    :[    7.319    ,    8.645    ],
                                8770    :[    7.327    ,    8.645    ],
                                8780    :[    7.335    ,    8.645    ],
                                8790    :[    7.343    ,    8.646    ],
                                8800    :[    7.351    ,    8.646    ],
                                8810    :[    7.359    ,    8.646    ],
                                8820    :[    7.367    ,    8.646    ],
                                8830    :[    7.375    ,    8.647    ],
                                8840    :[    7.383    ,    8.647    ],
                                8850    :[    7.391    ,    8.647    ],
                                8860    :[    7.399    ,    8.648    ],
                                8870    :[    7.407    ,    8.648    ],
                                8880    :[    7.415    ,    8.648    ],
                                8890    :[    7.423    ,    8.648    ],
                                8900    :[    7.432    ,    8.649    ],
                                8910    :[    7.440    ,    8.649    ],
                                8920    :[    7.448    ,    8.649    ],
                                8930    :[    7.456    ,    8.649    ],
                                8940    :[    7.464    ,    8.650    ],
                                8950    :[    7.472    ,    8.650    ],
                                8960    :[    7.480    ,    8.650    ],
                                8970    :[    7.488    ,    8.650    ],
                                8980    :[    7.496    ,    8.651    ],
                                8990    :[    7.504    ,    8.651    ],
                                9000    :[    7.512    ,    8.651    ]}


    if gas == "hydrogen":
        SpecEn = (33.3+39.4)/2    # specific energy (kWh/kg).  Using average if LHV and HHV due to blended use
        properties = Hydrogen properties
    elif gas == "natural gas":
        SpecEn = 13.9             # specific energy (kWh/kg)
        properties = Natural gas properties
    elif gas =="80/20 mix":
        print("sorry, the 80/20 mix is not yet available, please choose methane or hydrogen for the time
being")
        print("sorry, the 80/20 mix is not yet available, please choose methane or hydrogen for the time
being", file=fileout)
```

```
    else:
        print("Please identify the gas.  Check capitals.")
        print("Please identify the gas.  Check capitals.", file=fileout)

    # Define universal constants
    g = 9.81 # gravity (N/kg)
    pi = 3.1416 # pi (Mmm, pie)

    print("------ Section 2. Initial Pressures.  ------", file=fileout)
    #
    # Define initial Pressure at inlet node (kPa)
    InPres_NS00_Sfeg = 8500.0 # ie 85 bar, max in transmission mains as discussed with SGN
    print("", file=fileout)
    print("Initial Pressure at Sfeg inlet", file = fileout)
    print("InPres_NS00_Sfeg: ", InPres_NS00_Sfeg, file=fileout)
    print("",file=fileout)

    # Define demand profiles (MWh/h, hourly for 1 average day)
    # Demand nodes only, obviously.
    Profile_ND26_Wlot = [ 92.72, 61.81, 61.81, 61.81, 71.09, 120.54, 268.89, 309.07, 247.26, 216.35, 200.90,
200.90, 200.90, 200.90, 200.90, 253.44, 278.17, 293.62, 247.26, 231.81, 216.35, 154.54, 123.63, 92.72, ]
    Profile_ND27_Edin = [ 265.83, 177.22, 177.22, 177.22, 203.80, 345.57, 770.89, 886.08, 708.87, 620.26,
575.95, 575.95, 575.95, 575.95, 575.95, 726.59, 797.48, 841.78, 708.87, 664.56, 620.26, 443.04, 354.43,
265.83, ]
    Profile_ND28_Mlot = [ 46.82, 31.21, 31.21, 31.21, 35.90, 60.87, 135.78, 156.07, 124.86, 109.25, 101.45,
101.45, 101.45, 101.45, 101.45, 127.98, 140.47, 148.27, 124.86, 117.05, 109.25, 78.04, 62.43, 46.82, ]
    Profile_ND29_Elot = [ 54.23, 36.15, 36.15, 36.15, 41.58, 70.50, 157.27, 180.77, 144.61, 126.54, 117.50,
117.50, 117.50, 117.50, 117.50, 148.23, 162.69, 171.73, 144.61, 135.58, 126.54, 90.38, 72.31, 54.23, ]
    Profile_ND31_Fife = [ 189.17, 126.11, 126.11, 126.11, 145.03, 245.92, 548.58, 630.55, 504.44, 441.39,
409.86, 409.86, 409.86, 409.86, 409.86, 517.05, 567.50, 599.03, 504.44, 472.92, 441.39, 315.28, 252.22,
189.17, ]
    Profile_ND32_Dund = [ 75.62, 50.41, 50.41, 50.41, 57.97, 98.30, 219.29, 252.05, 201.64, 176.44, 163.83,
163.83, 163.83, 163.83, 163.83, 206.68, 226.85, 239.45, 201.64, 189.04, 176.44, 126.03, 100.82, 75.62, ]
    Profile_ND33_Angs = [ 58.84, 39.23, 39.23, 39.23, 45.11, 76.50, 170.65, 196.15, 156.92, 137.30, 127.50,
127.50, 127.50, 127.50, 127.50, 160.84, 176.53, 186.34, 156.92, 147.11, 137.30, 98.07, 78.46, 58.84, ]
    Profile_ND34_Absh = [ 132.28, 88.18, 88.18, 88.18, 101.41, 171.96, 383.60, 440.92, 352.74, 308.65, 286.60,
286.60, 286.60, 286.60, 286.60, 361.56, 396.83, 418.88, 352.74, 330.69, 308.65, 220.46, 176.37, 132.28, ]
    Profile_ND35_Abdn = [ 115.80, 77.20, 77.20, 77.20, 88.78, 150.54, 335.82, 386.00, 308.80, 270.20, 250.90,
250.90, 250.90, 250.90, 250.90, 316.52, 347.40, 366.70, 308.80, 289.50, 270.20, 193.00, 154.40, 115.80, ]
    Profile_ND36_Mory = [ 48.52, 32.35, 32.35, 32.35, 37.20, 63.08, 140.72, 161.74, 129.40, 113.22, 105.13,
105.13, 105.13, 105.13, 105.13, 132.63, 145.57, 153.66, 129.40, 121.31, 113.22, 80.87, 64.70, 48.52, ]
    Profile_ND37_High = [ 119.42, 79.62, 79.62, 79.62, 91.56, 155.25, 346.33, 398.08, 318.47, 278.66, 258.75,
258.75, 258.75, 258.75, 258.75, 326.43, 358.27, 378.18, 318.47, 298.56, 278.66, 199.04, 159.23, 119.42, ]
    Profile_ND39_Pkin = [ 76.95, 51.30, 51.30, 51.30, 58.99, 100.03, 223.15, 256.49, 205.19, 179.54, 166.72,
166.72, 166.72, 166.72, 166.72, 210.32, 230.84, 243.67, 205.19, 192.37, 179.54, 128.25, 102.60, 76.95, ]
    Profile_ND46_Sbor = [ 58.49, 39.00, 39.00, 39.00, 44.85, 76.04, 169.63, 194.98, 155.99, 136.49, 126.74,
126.74, 126.74, 126.74, 126.74, 159.88, 175.48, 185.23, 155.99, 146.24, 136.49, 97.49, 77.99, 58.49, ]

    print(" ------Section 3. Calculate flows ---------", file=fileout)
    print("Energy flux rates", file = fileout)
    #
    # Define arrays for energy flux rates
    # Network pipes
    Flux_PN00_Sfeg_Petc = []
    Flux_PN01_Petc_Foch = []
    Flux_PN02_Petc_Kiri = []
    Flux_PN03_Kiri_Glen = []
    Flux_PN04_Glen_Dunf = []
    Flux_PN05_Dunf_Balo = []
    Flux_PN06_Balo_Laud = []
    #Flux_PN07_Laud_Cold = []

    #   Demand pipes
    Flux_PD19_Petc_Abdn = []
    Flux_PD20_Petc_Absh = []
    Flux_PD21_Foch_Mory = []
    Flux_PD22_Foch_High = []
    Flux_PD23_Kiri_Angs = []
    Flux_PD24_Glen_Dund = []
    Flux_PD25_Glen_Pkin = []
    Flux_PD26_Dunf_Fife = []
    Flux_PD27_Balo_Wlot = []
    Flux_PD28_Balo_Edin = []
    Flux_PD29_Balo_Mlot = []
```

```
    Flux_PD30_Laud_Elot = []
    Flux_PD31_Laud_Sbor = []

    # Calculate Energy Flux rate in pipes (kWh/sec) due to demand
    # 1000 factor converts MWh to kWh
    # 3600 factor converts Flux per hour to Flux per second
    # Demand pipes entered first as network pipes use those as inputs - so this basically counts back up the
network.
    # Also Network pipes are entered in reverse order, for the same reason.
    # Define counter for hours
    Hours = list(range(0,24))

    for h in Hours:
        #   Demand pipes
        Flux_PD19_Petc_Abdn_h = (Profile_ND35_Abdn[h] * 1000 /3600)
        Flux_PD20_Petc_Absh_h = (Profile_ND34_Absh[h] * 1000 /3600)
        Flux_PD21_Foch_Mory_h = (Profile_ND36_Mory[h] * 1000 /3600)
        Flux_PD22_Foch_High_h = (Profile_ND37_High[h] * 1000 /3600)
        Flux_PD23_Kiri_Angs_h = (Profile_ND33_Angs[h] * 1000 /3600)
        Flux_PD24_Glen_Dund_h = (Profile_ND32_Dund[h] * 1000 /3600)
        Flux_PD25_Glen_Pkin_h = (Profile_ND39_Pkin[h] * 1000 /3600)
        Flux_PD26_Dunf_Fife_h = (Profile_ND31_Fife[h] * 1000 /3600)
        Flux_PD27_Balo_Wlot_h = (Profile_ND26_Wlot[h] * 1000 /3600)
        Flux_PD28_Balo_Edin_h = (Profile_ND27_Edin[h] * 1000 /3600)
        Flux_PD29_Balo_Mlot_h = (Profile_ND28_Mlot[h] * 1000 /3600)
        Flux_PD30_Laud_Elot_h = (Profile_ND29_Elot[h] * 1000 /3600)
        Flux_PD31_Laud_Sbor_h = (Profile_ND46_Sbor[h] * 1000 /3600)

        # Network pipes (reverse order)
        Flux_PN07_Laud_Cold_h =  0
        Flux_PN06_Balo_Laud_h = (Flux_PN07_Laud_Cold_h + Flux_PD30_Laud_Elot_h + Flux_PD31_Laud_Sbor_h)
        Flux_PN05_Dunf_Balo_h = (Flux_PN06_Balo_Laud_h + Flux_PD27_Balo_Wlot_h + Flux_PD28_Balo_Edin_h +
Flux_PD29_Balo_Mlot_h)
        Flux_PN04_Glen_Dunf_h = (Flux_PN05_Dunf_Balo_h + Flux_PD26_Dunf_Fife_h)
        Flux_PN03_Kiri_Glen_h = (Flux_PN04_Glen_Dunf_h + Flux_PD25_Glen_Pkin_h + Flux_PD24_Glen_Dund_h)
        Flux_PN02_Petc_Kiri_h = (Flux_PN03_Kiri_Glen_h + Flux_PD23_Kiri_Angs_h)
        Flux_PN01_Petc_Foch_h = (Flux_PD21_Foch_Mory_h + Flux_PD22_Foch_High_h)
        Flux_PN00_Sfeg_Petc_h = (Flux_PN01_Petc_Foch_h + Flux_PN02_Petc_Kiri_h + Flux_PD19_Petc_Abdn_h +
Flux_PD20_Petc_Absh_h)

        #   Append Flux array
        #   Network pipes
        Flux_PN00_Sfeg_Petc.append(Flux_PN00_Sfeg_Petc_h)
        Flux_PN01_Petc_Foch.append(Flux_PN01_Petc_Foch_h)
        Flux_PN02_Petc_Kiri.append(Flux_PN02_Petc_Kiri_h)
        Flux_PN03_Kiri_Glen.append(Flux_PN03_Kiri_Glen_h)
        Flux_PN04_Glen_Dunf.append(Flux_PN04_Glen_Dunf_h)
        Flux_PN05_Dunf_Balo.append(Flux_PN05_Dunf_Balo_h)
        Flux_PN06_Balo_Laud.append(Flux_PN06_Balo_Laud_h)
       # Flux_PN07_Laud_Cold.append(Flux_PN07_Laud_Cold_h)

        #   Demand pipes
        Flux_PD19_Petc_Abdn.append(Flux_PD19_Petc_Abdn_h)
        Flux_PD20_Petc_Absh.append(Flux_PD20_Petc_Absh_h)
        Flux_PD21_Foch_Mory.append(Flux_PD21_Foch_Mory_h)
        Flux_PD22_Foch_High.append(Flux_PD22_Foch_High_h)
        Flux_PD23_Kiri_Angs.append(Flux_PD23_Kiri_Angs_h)
        Flux_PD24_Glen_Dund.append(Flux_PD24_Glen_Dund_h)
        Flux_PD25_Glen_Pkin.append(Flux_PD25_Glen_Pkin_h)
        Flux_PD26_Dunf_Fife.append(Flux_PD26_Dunf_Fife_h)
        Flux_PD27_Balo_Wlot.append(Flux_PD27_Balo_Wlot_h)
        Flux_PD28_Balo_Edin.append(Flux_PD28_Balo_Edin_h)
        Flux_PD29_Balo_Mlot.append(Flux_PD29_Balo_Mlot_h)
        Flux_PD30_Laud_Elot.append(Flux_PD30_Laud_Elot_h)
        Flux_PD31_Laud_Sbor.append(Flux_PD31_Laud_Sbor_h)

    # print Flux rates
    # print header for output
    print("Calculated energy Flux rates for each pipe. kWh/s.  Network Pipes", file = fileout)
    print("{:<5} {:<14} {:<14} {:<14} {:<14} {:<14} {:<14} ".\
        format("Time", "PN00_Sfrg_Petc", "PN01_Petc_Foch", "PN02_Petc_Kiri", "PN03_Kiri_Glen",
"PN04_Glen_Dunf", "PN05_Dunf_Balo", "PN06_Balo_Laud"), file=fileout)
    # print("", file=fileout)
    for h in Hours:
```

```
        print("{:<5} {:<14} {:<14} {:<14} {:<14} {:<14} {:<14} ".\
            format(h, Flux_PN00_Sfeg_Petc[h], Flux_PN01_Petc_Foch[h], Flux_PN02_Petc_Kiri[h],
Flux_PN03_Kiri_Glen[h], Flux_PN04_Glen_Dunf[h], Flux_PN05_Dunf_Balo[h],\
                    Flux_PN06_Balo_Laud[h], ),file=fileout)
    print("",file=fileout)

    print("Calculated energy Flux rates for each pipe. kWh/s.  Demand Pipes", file = fileout)
    print("{:<5} {:<14} {:<14} {:<14} {:<14} {:<14} {:<14} {:<14} {:<14} {:<14} {:<14} {:<14}".\
        format("Time", "PD19_Petc_Abdn", "PD20_Petc_Absh", "PD21_Foch_Mory", "PD22_Foch_High",
"PD23_Kiri_Angs", "PD24_Glen_Dund", "PD25_Glen_Pkin", "PD26_Dunf_Fife",\
              "PD27_Balo_Wlot", "PD28_Balo_Edin", "PD29_Balo_Mlot", "PD30_Laud_Elot", "PD31_Laud_Sbor",),
file=fileout)
    # print("", file=fileout)
    for h in Hours:
        print("{:<5} {:<14} {:<14} {:<14} {:<14} {:<14} {:<14} {:<14} {:<14} {:<14} {:<14} {:<14}
{:<14} ".\
            format(h, Flux_PD19_Petc_Abdn[h], Flux_PD20_Petc_Absh[h], Flux_PD21_Foch_Mory[h],
Flux_PD22_Foch_High[h], Flux_PD23_Kiri_Angs[h], Flux_PD24_Glen_Dund[h],\
                    Flux_PD25_Glen_Pkin[h], Flux_PD26_Dunf_Fife[h], Flux_PD27_Balo_Wlot[h],
Flux_PD28_Balo_Edin[h], Flux_PD29_Balo_Mlot[h], Flux_PD30_Laud_Elot[h],\
                     Flux_PD31_Laud_Sbor[h]),file=fileout)
    print("",file=fileout)

    #Calculate mass flow rates
    print("Mass flow rates", file = fileout)

    # Define arrays for mass flow rates
    # Network pipes
    Mflo_PN00_Sfeg_Petc = []
    Mflo_PN01_Petc_Foch = []
    Mflo_PN02_Petc_Kiri = []
    Mflo_PN03_Kiri_Glen = []
    Mflo_PN04_Glen_Dunf = []
    Mflo_PN05_Dunf_Balo = []
    Mflo_PN06_Balo_Laud = []
    #Mflo_PN07_Laud_Cold = []

    #   Demand pipes
    Mflo_PD19_Petc_Abdn = []
    Mflo_PD20_Petc_Absh = []
    Mflo_PD21_Foch_Mory = []
    Mflo_PD22_Foch_High = []
    Mflo_PD23_Kiri_Angs = []
    Mflo_PD24_Glen_Dund = []
    Mflo_PD25_Glen_Pkin = []
    Mflo_PD26_Dunf_Fife = []
    Mflo_PD27_Balo_Wlot = []
    Mflo_PD28_Balo_Edin = []
    Mflo_PD29_Balo_Mlot = []
    Mflo_PD30_Laud_Elot = []
    Mflo_PD31_Laud_Sbor = []

    # Calculate Mass flow rate in pipes (kg/hour) due to demand
    # 1000 factor converts MWh to kWh
    # SpecEn factor converts energy flux per hour to mass flow per hour
    # Demand pipes entered first as network pipes use those as inputs - so this basically counts back up the
network.
    # Also Network pipes are entered in reverse order, for the same reason.

    for h in Hours:
        #   Demand pipes
        MFlo_PD19_Petc_Abdn_h = (Profile_ND35_Abdn[h] * 1000 / SpecEn)
        Mflo_PD20_Petc_Absh_h = (Profile_ND34_Absh[h] * 1000 / SpecEn)
        Mflo_PD21_Foch_Mory_h = (Profile_ND36_Mory[h] * 1000 / SpecEn)
        Mflo_PD22_Foch_High_h = (Profile_ND37_High[h] * 1000 / SpecEn)
        Mflo_PD23_Kiri_Angs_h = (Profile_ND33_Angs[h] * 1000 / SpecEn)
        Mflo_PD24_Glen_Dund_h = (Profile_ND32_Dund[h] * 1000 / SpecEn)
        Mflo_PD25_Glen_Pkin_h = (Profile_ND39_Pkin[h] * 1000 / SpecEn)
        Mflo_PD26_Dunf_Fife_h = (Profile_ND31_Fife[h] * 1000 / SpecEn)
        Mflo_PD27_Balo_Wlot_h = (Profile_ND26_Wlot[h] * 1000 / SpecEn)
        Mflo_PD28_Balo_Edin_h = (Profile_ND27_Edin[h] * 1000 / SpecEn)
        Mflo_PD29_Balo_Mlot_h = (Profile_ND28_Mlot[h] * 1000 / SpecEn)
        Mflo_PD30_Laud_Elot_h = (Profile_ND29_Elot[h] * 1000 / SpecEn)
        Mflo_PD31_Laud_Sbor_h = (Profile_ND46_Sbor[h] * 1000 / SpecEn)
```

```
        # Network pipes (reverse order)
        Mflo_PN07_Laud_Cold_h =  0
        Mflo_PN06_Balo_Laud_h = (Mflo_PN07_Laud_Cold_h + Mflo_PD30_Laud_Elot_h + Mflo_PD31_Laud_Sbor_h)
        Mflo_PN05_Dunf_Balo_h = (Mflo_PN06_Balo_Laud_h + Mflo_PD27_Balo_Wlot_h + Mflo_PD28_Balo_Edin_h +
Mflo_PD29_Balo_Mlot_h)
        Mflo_PN04_Glen_Dunf_h = (Mflo_PN05_Dunf_Balo_h + Mflo_PD26_Dunf_Fife_h)
        Mflo_PN03_Kiri_Glen_h = (Mflo_PN04_Glen_Dunf_h + Mflo_PD25_Glen_Pkin_h + Mflo_PD24_Glen_Dund_h)
        Mflo_PN02_Petc_Kiri_h = (Mflo_PN03_Kiri_Glen_h + Mflo_PD23_Kiri_Angs_h)
        Mflo_PN01_Petc_Foch_h = (Mflo_PD21_Foch_Mory_h + Mflo_PD22_Foch_High_h)
        Mflo_PN00_Sfeg_Petc_h = (Mflo_PN01_Petc_Foch_h + Mflo_PN02_Petc_Kiri_h + MFlo_PD19_Petc_Abdn_h +
Mflo_PD20_Petc_Absh_h)

        #   Append Mass flow array
        #   Network pipes
        Mflo_PN00_Sfeg_Petc.append(Mflo_PN00_Sfeg_Petc_h)
        Mflo_PN01_Petc_Foch.append(Mflo_PN01_Petc_Foch_h)
        Mflo_PN02_Petc_Kiri.append(Mflo_PN02_Petc_Kiri_h)
        Mflo_PN03_Kiri_Glen.append(Mflo_PN03_Kiri_Glen_h)
        Mflo_PN04_Glen_Dunf.append(Mflo_PN04_Glen_Dunf_h)
        Mflo_PN05_Dunf_Balo.append(Mflo_PN05_Dunf_Balo_h)
        Mflo_PN06_Balo_Laud.append(Mflo_PN06_Balo_Laud_h)
       # Mflo_PN07_Laud_Cold.append(Mflo_PN07_Laud_Cold_h)

        #   Demand pipes
        Mflo_PD19_Petc_Abdn.append(MFlo_PD19_Petc_Abdn_h)
        Mflo_PD20_Petc_Absh.append(Mflo_PD20_Petc_Absh_h)
        Mflo_PD21_Foch_Mory.append(Mflo_PD21_Foch_Mory_h)
        Mflo_PD22_Foch_High.append(Mflo_PD22_Foch_High_h)
        Mflo_PD23_Kiri_Angs.append(Mflo_PD23_Kiri_Angs_h)
        Mflo_PD24_Glen_Dund.append(Mflo_PD24_Glen_Dund_h)
        Mflo_PD25_Glen_Pkin.append(Mflo_PD25_Glen_Pkin_h)
        Mflo_PD26_Dunf_Fife.append(Mflo_PD26_Dunf_Fife_h)
        Mflo_PD27_Balo_Wlot.append(Mflo_PD27_Balo_Wlot_h)
        Mflo_PD28_Balo_Edin.append(Mflo_PD28_Balo_Edin_h)
        Mflo_PD29_Balo_Mlot.append(Mflo_PD29_Balo_Mlot_h)
        Mflo_PD30_Laud_Elot.append(Mflo_PD30_Laud_Elot_h)
        Mflo_PD31_Laud_Sbor.append(Mflo_PD31_Laud_Sbor_h)

    # print mass flow rates
    # print header for output
    print("Calculated mass flow rates for each pipe. kg/hr.  Network Pipes", file = fileout)
    print("{:<5} {:<14} {:<14} {:<14} {:<14} {:<14} {:<14} {:<14} ".\
        format("Time", "PN00 Sfrg Petc", "PN01 Petc Foch", "PN02 Petc Kiri", "PN03 Kiri Glen",
"PN04 Glen Dunf", "PN05 Dunf Balo", "PN06 Balo Laud"), file=fileout)
    # print("", file=fileout)
    for h in Hours:
        print("{:<5} {:<14,} {:<14,} {:<14,} {:<14,} {:<14,} {:<14,} {:<14,} ".\
            format(h, round(Mflo_PN00_Sfeg_Petc[h]), round(Mflo_PN01_Petc_Foch[h]),
round(Mflo_PN02_Petc_Kiri[h]), round(Mflo_PN03_Kiri_Glen[h]), round(Mflo_PN04_Glen_Dunf[h]),
round(Mflo_PN05_Dunf_Balo[h]), round(Mflo_PN06_Balo_Laud[h]), ),file=fileout)
    print("",file=fileout)

    print("Calculated mass flow rates for each pipe. kg/hr.  Demand Pipes", file = fileout)
    print("{:<5} {:<14} {:<14} {:<14} {:<14} {:<14} {:<14} {:<14} {:<14} {:<14} {:<14} {:<14} {:<14} {:<14}".\
        format("Time", "PD19 Petc Abdn", "PD20 Petc Absh", "PD21 Foch Mory", "PD22 Foch High",
"PD23 Kiri Angs", "PD24 Glen Dund", "PD25 Glen Pkin", "PD26 Dunf Fife", "PD27 Balo Wlot", "PD28 Balo Edin",
"PD29 Balo Mlot", "PD30 Laud Elot", "PD31 Laud Sbor",), file=fileout)
    # print("", file=fileout)
    for h in Hours:
        print("{:<5} {:<14,} {:<14,} {:<14,} {:<14,} {:<14,} {:<14,} {:<14,} {:<14,} {:<14,} {:<14,} {:<14,}
{:<14,} {:<14,} ".\
            format(h, round(Mflo_PD19_Petc_Abdn[h]), round(Mflo_PD20_Petc_Absh[h]),
round(Mflo_PD21_Foch_Mory[h]), round(Mflo_PD22_Foch_High[h]), round(Mflo_PD23_Kiri_Angs[h]),
round(Mflo_PD24_Glen_Dund[h]), round(Mflo_PD25_Glen_Pkin[h]), round(Mflo_PD26_Dunf_Fife[h]),
round(Mflo_PD27_Balo_Wlot[h]), round(Mflo_PD28_Balo_Edin[h]), round(Mflo_PD29_Balo_Mlot[h]),
round(Mflo_PD30_Laud_Elot[h]), round(Mflo_PD31_Laud_Sbor[h])),file=fileout)
    print("",file=fileout)

    print(" ------Section 4. Calculate Pressures ---------", file=fileout)

    #   Pressure drops - Network
    PresDrop_PN00_Sfeg_Petc = []
    PresDrop_PN01_Petc_Foch = []
```

```
    PresDrop_PN02_Petc_Kiri = []
    PresDrop_PN03_Kiri_Glen = []
    PresDrop_PN04_Glen_Dunf = []
    PresDrop_PN05_Dunf_Balo = []
    PresDrop_PN06_Balo_Laud = []
    #PresDrop_PN07_Laud_Cold = []

    #   Pressure drops - Demand
    PresDrop_PD19_Petc_Abdn = []
    PresDrop_PD20_Petc_Absh = []
    PresDrop_PD21_Foch_Mory = []
    PresDrop_PD22_Foch_High = []
    PresDrop_PD23_Kiri_Angs = []
    PresDrop_PD24_Glen_Dund = []
    PresDrop_PD25_Glen_Pkin = []
    PresDrop_PD26_Dunf_Fife = []
    PresDrop_PD27_Balo_Wlot = []
    PresDrop_PD28_Balo_Edin = []
    PresDrop_PD29_Balo_Mlot = []
    PresDrop_PD30_Laud_Elot = []
    PresDrop_PD31_Laud_Sbor = []
    #
    #
    # ----------------- Buzzelli (2008) & Darcy-Weisbach Calculations -define PressureDrop function ----------
------- #

    # Define function to calculate Pressure drop using Buzzelli equation
    # inputs: (D,Qe,ks,L,Pup,props,name,h, minpress)
    # D = pipe diameter (m)
    # Qe = energy Flux (kWh/s)
    # ks = pipe roughness (mm)
    # L = pipe length (m)
    # Pup (kPa)
    # props = properties (relevant dictionary above)
    # name = pipe name
    # h = time increment
    # minpress = minimum allowable downstream pressure, (kPa)

    def PressureDrop(D,Qe,ks,L,Pup,props,name,h, minpress):
        # Carries out calculation from upstream pressure down to zero downstream pressure, until input
pressure drop (Pdrop) = calculated Pdrop (PdropC)

        # tests for upstream pressure.  If zero or negative, retuns error.  Otherwise continues calculation.
        NoPup = 0       # Sets up error counter for zero or negative upstream pressure
        noconverge = 0  # Sets up error counter for non-converging iterations
        zerodrop = 0    # Sets up error counter for zero pressure drop, ie zero flow
        lowpressure = 0 # Sets up error counter for inadequate upstream pressure for flow demanded. Max
available flow is returned

        if Pup <= 0:
            print("Timestep",h, name, "Upstream pressure zero or negative.  No calculations. Pdrop, Vel, rho,
mu, linepack all set to zero.", file = fileout)
            Pdrop = 0
            Pdown = Pup
            Vel = 0
            Rho = 0
            Mu = 0
            Linepack = 0
            NoPup = 1
            QMcalc = 0

        else:
            Pdrop1 = 1      # kPa    # intial guess of Pressure drop for iterative calculation
            count = 1
            maxcount = 200      # Limits number of iterations

            while count <=maxcount:   # iteration counter

                # Calculates downstream pressure and average pressure based
                Pdown = Pup - Pdrop1  #kPa
                Pav = round((Pup + Pdown)/20)*10  #kPa

                if Pav<=9000 and Pav >= 10:  #kPa
                    rho = props[Pav][0]      # kg/m3
```

```
                    mu = props[Pav][1]/1000000   #Pa-s
                elif Pav >9000:             #kPa
                    rho = props[9000][0]   #kg/m3
                    mu = props[9000][1]/1000000   #Pa-s
                else: # ie Pav>0 and <10 - use values for 10.
                    rho = props[10][0]          #kg/m3
                    mu = props[10][1]/1000000   #Pa-s

                # Calculates flow velocity, V
                V = Qe/(SpecEn*rho*(pi*D**2/4))   # V (m/s) = Energy flux (kWh/s) / [Specific energy (kWh/kg)
x Density (kg/m3) x Pipe Area (m2)]

                # Calculates Reynolds Number, Re.  Re = D.V.rho/mu
                Re = D*V*rho/mu  # m * m/s * kg/m3 / (kg/ms) = dimless

                # Calculates Buzzelli component B1.  B1 = (0.774.ln(Re)-1.41) / (1+1.32.sqrt(ks/D))
                B1 = (0.774*math.log(Re) - 1.41) / (1 + 1.32* ((ks/1000)/D)**0.5)

                # Calculates Buzzelli component B2.  B2 = ks.Re/3.7D + 2.51.B1
                B2 = (ks/1000)*Re/(3.7*D) + 2.51*B1

                # Calculates fD (aka Lambda) according to Buzzelli (2008).  1/sqrt(fD) = B1 - ( (B1 +
2.log10(B2/Re) ) / (1 + 2.18/B2) )
                fD = ( B1 - ( (B1 + 2*math.log10(B2/Re) ) / (1 + 2.18/B2) ) )**-2

                # Calculates pressure drop according to Darcy-Weisbach formula.  Pdrop = L*fD*rho*V**2/(2*D)
                Pdrop2 = (L*fD*rho*V**2/(2*D))/1000  # Factor of 1/1000 converts result in Pa to kPa

                # Calculates mass flow achieved
                QMcalc = rho * V * (pi*(D**2)/4) * 3600

                print("Timestep ",h, "Iteration ",count," ",name,":
D",D,"m.  V",round(V,3),"m/s.  ks",round(ks,7),"mm.  Rho ",round(rho,4)," kg/m3.  mu ",\
                      round(mu,7),"Pa-s. Pup ",round(Pup,2)," kPa. Pdrop1(input) ",round(Pdrop1,2), " kPa.
Pdrop2 (output)",round(Pdrop2,2),"kPa.", file=fileout)

                print("Timestep ",h, "Iteration ",count," ",name,": Re: ",round(Re), ". B1: ",round(B1,2),".
B2: ",round(B2,2),". fD:", round(fD,5),"Qe",round(Qe,2),\
                      "SpecEn",round(SpecEn,2),"kWh/kg. L",round(L),"m. rho",rho,"kg/m3. Qm",QMcalc,"kg/hour",
file=fileout)

                print("",file = fileout)

                # Iterate calculation counter
                count = count + 1

                if Pdrop2 > (Pup-minpress):
                    print("negative pressure after ", count, "iterations.  Downstream pressure set to minimum
allowable for pipe type (network or demand). Restricted downstream flow calculated.",
 file=fileout)
                    # Inadequate pressure error.  Impossible to push all the gas demanded down the pipe with
the pressure available.  Computes the flow available.
                    # Calculate available flow for Pdrop = Pup-minpress and Pdown = minpress.
                    # Break.
                    cnt = 1
                    maxcnt = 200
                    Pdrop = Pup                 # kPa
                    Pdown = minpress            # kPa
                    Pav = round((Pup+Pdown)/20)*10 # kPa.  Function rounds result to nearest 10kPa
                    rho = props[Pav][0]         # kg/m3
                    mu = props[Pav][1]/1000000  # Pa.s or kg/m.s
                    Vcalc1 = 1                  # m/s

                    while cnt < maxcnt:         # NB Inadequate pressure output calculator
                        Re = (D)*Vcalc1*rho/mu                                          # dimless
                        B1 = (0.774*math.log(Re) - 1.41) / (1 + 1.32*((ks/1000)/D)**0.5)    # dimless
                        B2 = (ks/1000)*Re/(3.7*D) + 2.51*B1                             # dimless
                        fD = ( B1 - ( (B1 + 2*math.log10(B2/Re) ) / (1 + 2.18/B2) ) )**-2   # IDFK
                        Vcalc2 = ( (2*D*(Pdrop*1000)) / (L*fD*rho) )**0.5               # m/s

                        if Vcalc2 <= Vcalc1 * 1.001 and Vcalc2 >= 0.999 * Vcalc1:
# Accept calculated Vcalc2 if within 0.1% of Vcalc1.
                            Pdrop = Pup - minpress         # kPa
                            Pdown = minpress               # kPa
```

```
                        Vel = Vcalc2                    # m/s
                        Rho = rho                       # kg/m3
                        Mu = mu                         # Pa.s
                        lowpressure = 1                 # NB Inadequate pressure counter
                        QMcalc = rho * Vel * (pi*(D**2)/4) * 3600   # Calculates actual mass flow
                        Linepack = rho * (pi*(D**2)/4) * L   # Calculates mass of gas contained in pipe
 length.  Density * Area * Length
                        print("Final, restricted: Pipe: ",name,  h,"| Pup: ",round(Pup,2)," Pdrop: ",
round(Pdrop,2),"kPa. Pdown: ",round(Pdown,2), "kPa. Rho: ",rho," kg/m3. Mu: ",mu * 1000000," uPa-s. Re:
",round(Re), file=fileout)

                        print("B1: ",round(B1,2),"  B2: ",round(B2,2)," fD:", round(fD,4),"| Velocity:
",round(Vel,3)," m/s. Mass flow: ",QMcalc,"kg/hour.  Linepack: ",round(Linepack),"kg.  Not resolved,
restricted flow = ", round(Vel * D**2*pi*0.25 * Rho * 3600), " kg/hr", file=fileout)

                        print("",file = fileout)
                        print("",file = fileout)
                        break

                    if cnt == maxcnt:          # NB Inadequate pressure output calculator
                        print("restricted flow did not converge after ", cnt, "iterations.  Downstream
pressure and flow set to zero.",file=fileout)
                        print("",file=fileout)
                        Pdrop = Pup  #kPa
                        Pdown = 0    #kPa
                        Vel = 0      #m/s
                        Rho = rho    #kg/m3
                        Mu = mu      #Pa.s
                        Linepack = 0
                        QMcalc = 0
                        lowpressure = 1  # Inadequate pressure counter
                        noconverge = 1   # Non-converging calculation counter
                        break

                    # Iterate estimated V and counter within inadequate pressure output calculator
                    Vcalc1 = Vcalc1 + (Vcalc2 - Vcalc1)/10   # m/s
                    cnt=cnt+1
                break

            if V == 0:
                # Zero flow error.
                # No calculations.
                # Break.
                print("Zero pressure drop", file = fileout)
                print("", file=fileout)
                Pdrop = 0
                Pdown = Pup
                Vel = 0
                Rho = rho
                Mu = mu
                QMcalc = 0
                Linepack = 0
                zerodrop = 1
                break

            if count == maxcount:
                # Did Not Converge error.
                # No calculations.
                # Break.
                print("did not converge after ", count, "iterations.  Pressure drop and flow set to
zero",file=fileout)
                print("",file=fileout)
                Pdrop = Pup
                Pdown = 0
                Vel = 0
                Rho = rho
                Mu = mu
                QMcalc = 0
                Linepack = 0
                noconverge = 1
                break

            # If output and input Pdrops are close enough, accept result and stop iterating.
            if Pdrop2 <= Pdrop1 * 1.001 and Pdrop2 >= 0.999 * Pdrop1:
```

```
                    Pdrop = Pdrop2
                    Pdown = Pup - Pdrop
                    Vel = V
                    Rho = rho
                    Mu = mu
                    QMcalc = rho * Vel * (pi*(D**2)/4) * 3600
                    Linepack = rho * (pi*(D**2)/4) * L    # Calculates mass of gas contained in pipe
length.  Density * Area * Length
                    print("Final: Pipe: ",name,  h,"| Pup: ",round(Pup,2)," Pdrop: ", round(Pdrop,2),"kPa.
Pdown: ",round(Pdown,2), "kPa. Rho: ",rho,\
                          "kg/m3. Mu: ",mu * 1000000," uPa-s. Re: ",round(Re),"Qmcalc: ",QMcalc,"kg/hour.",
 file=fileout)

                    print("Final: Pipe: ",name,  h,"| B1: ",round(B1,2),"  B2: ",round(B2,2)," fD:",
round(fD,4),"| Velocity: ",round(Vel,3)," m/s. Linepack: ",\
                          round(Linepack),"kg. Resolved at ",count," iterations.", file=fileout)

                    print("",file=fileout)
                    print("",file=fileout)
                    break

                # Iterate calculation.
                # Find new Pdrop to evaluate. Next iteration to try is 1/10 of average of Pdrop1 and Pdrop2
past Pdrop1.
                # This resulted in a quicker computation time than the simple average.
                Pdrop1 = Pdrop1 + (Pdrop2 - Pdrop1)/10  # kPa

        # Return outputs from calculation
        return(Pdrop, Pdown, Vel, Rho, Mu, noconverge, zerodrop, lowpressure, NoPup, Linepack, QMcalc)



    ##------------  Calculate Pressure drop in pipes using Buzzelli & Darcy-Weisbach equations [using
PressureDrop function above] -----------##

    #   Set up input parameters:
(D,Qe,ks,L,props,name,h)
    #   Call function "PressureDrop" to calculate Pressure Drop for each set of inputs

    NoCon = 0 # error check - counts total pipes with no convergence.  For valid results, must equal zero.
    ZeroD = 0 # error check - counts total pipes with no pressure drop.  For valid results, must equal zero.
    LowPress = 0 # error check - counts pipes with inadequate upstream pressure.  For valid results must = 0.
    ZeroPup = 0 # error check - counts pipes with zero or negative pressure upstream.
    Lpacknet = [] # Sum of linepack in network pipes (kg)
    Lpackdem = [] # Sum of linepack in demand pipes (kg)
    Lpacktot = [] # Total linepack (kg)

    for h in Hours:
        print("",file=fileout)

        print("timestep start: ", h, file=fileout)
        # Network
        PresDrop_PN00_Sfeg_Petc_h = PressureDrop(Param_PN00_Sfeg_Petc[0], Flux_PN00_Sfeg_Petc[h],
Param_PN00_Sfeg_Petc[1], Len_PN00_Sfeg_Petc, \
                                                 InPres_NS00_Sfeg, properties, "PN00_Sfeg_Petc",h,
Param_PN00_Sfeg_Petc[2])
                                                 # D (m), Qe (kWh/s), ks (mm), L (m), Pup (kPa), Rho & Mu
table, name, timestep, minpress (kPa)
        PresDrop_PN00_Sfeg_Petc.append(PresDrop_PN00_Sfeg_Petc_h)
        NoCon = NoCon + PresDrop_PN00_Sfeg_Petc_h[5]
        ZeroD = ZeroD + PresDrop_PN00_Sfeg_Petc_h[6]
        LowPress = LowPress + PresDrop_PN00_Sfeg_Petc_h[7]
        ZeroPup = ZeroPup + PresDrop_PN00_Sfeg_Petc_h[8]
        Lpacknet.append(PresDrop_PN00_Sfeg_Petc_h[9])
        # print("Lpacknet[",h,"]: ",Lpacknet[h])  # Error checking

        PresDrop_PN01_Petc_Foch_h = PressureDrop(Param_PN01_Petc_Foch[0], Flux_PN01_Petc_Foch[h],
Param_PN01_Petc_Foch[1], Len_PN01_Petc_Foch, \
                                                 PresDrop_PN00_Sfeg_Petc[h][1], properties,
"PN01_Petc_Foch",h, Param_PN01_Petc_Foch[2])
        PresDrop_PN01_Petc_Foch.append(PresDrop_PN01_Petc_Foch_h)
        NoCon = NoCon + PresDrop_PN01_Petc_Foch_h[5]
```

```
        ZeroD = ZeroD + PresDrop_PN01_Petc_Foch_h[6]
        LowPress = LowPress + PresDrop_PN00_Sfeg_Petc_h[7]
        ZeroPup = ZeroPup + PresDrop_PN00_Sfeg_Petc_h[8]
        Lpacknet[h] = Lpacknet[h] + PresDrop_PN01_Petc_Foch_h[9]
        # print("Lpacknet[",h,"]: ",Lpacknet[h])

        PresDrop_PN02_Petc_Kiri_h = PressureDrop(Param_PN02_Petc_Kiri[0], Flux_PN02_Petc_Kiri[h],
Param_PN02_Petc_Kiri[1], Len_PN02_Petc_Kiri, \
                                                 PresDrop_PN00_Sfeg_Petc[h][1], properties,
"PN02_Petc_Kiri",h, Param_PN02_Petc_Kiri[2])
        PresDrop_PN02_Petc_Kiri.append(PresDrop_PN02_Petc_Kiri_h)
        NoCon = NoCon + PresDrop_PN02_Petc_Kiri_h[5]
        ZeroD = ZeroD + PresDrop_PN02_Petc_Kiri_h[6]
        LowPress = LowPress + PresDrop_PN02_Petc_Kiri_h[7]
        ZeroPup = ZeroPup + PresDrop_PN02_Petc_Kiri_h[8]
        Lpacknet[h] = Lpacknet[h] + PresDrop_PN02_Petc_Kiri_h[9]
        # print("Lpacknet[",h,"]: ",Lpacknet[h])

        PresDrop_PN03_Kiri_Glen_h = PressureDrop(Param_PN03_Kiri_Glen[0], Flux_PN03_Kiri_Glen[h],
Param_PN03_Kiri_Glen[1], Len_PN03_Kiri_Glen, \
                                                 PresDrop_PN02_Petc_Kiri[h][1], properties,
"PN03_Kiri_Glen",h, Param_PN03_Kiri_Glen[2])
        PresDrop_PN03_Kiri_Glen.append(PresDrop_PN03_Kiri_Glen_h)
        NoCon = NoCon + PresDrop_PN03_Kiri_Glen_h[5]
        ZeroD = ZeroD + PresDrop_PN03_Kiri_Glen_h[6]
        LowPress = LowPress + PresDrop_PN03_Kiri_Glen_h[7]
        ZeroPup = ZeroPup + PresDrop_PN03_Kiri_Glen_h[8]
        Lpacknet[h] = Lpacknet[h] + PresDrop_PN03_Kiri_Glen_h[9]
        # print("Lpacknet[",h,"]: ",Lpacknet[h])


        PresDrop_PN04_Glen_Dunf_h = PressureDrop(Param_PN04_Glen_Dunf[0], Flux_PN04_Glen_Dunf[h],
Param_PN04_Glen_Dunf[1], Len_PN04_Glen_Dunf, \
                                                 PresDrop_PN03_Kiri_Glen[h][1], properties,
"PN04_Glen_Dunf",h, Param_PN04_Glen_Dunf[2])
        PresDrop_PN04_Glen_Dunf.append(PresDrop_PN04_Glen_Dunf_h)
        NoCon = NoCon + PresDrop_PN04_Glen_Dunf_h[5]
        ZeroD = ZeroD + PresDrop_PN04_Glen_Dunf_h[6]
        LowPress = LowPress + PresDrop_PN04_Glen_Dunf_h[7]
        ZeroPup = ZeroPup + PresDrop_PN04_Glen_Dunf_h[8]
        Lpacknet[h] = Lpacknet[h] + PresDrop_PN04_Glen_Dunf_h[9]
        # print("Lpacknet[",h,"]: ",Lpacknet[h])

        PresDrop_PN05_Dunf_Balo_h = PressureDrop(Param_PN05_Dunf_Balo[0], Flux_PN05_Dunf_Balo[h],
Param_PN05_Dunf_Balo[1], Len_PN05_Dunf_Balo, \
                                                 PresDrop_PN04_Glen_Dunf[h][1], properties,
"PN05_Dunf_Balo",h, Param_PN05_Dunf_Balo[2])
        PresDrop_PN05_Dunf_Balo.append(PresDrop_PN05_Dunf_Balo_h)
        NoCon = NoCon + PresDrop_PN05_Dunf_Balo_h[5]
        ZeroD = ZeroD + PresDrop_PN05_Dunf_Balo_h[6]
        LowPress = LowPress + PresDrop_PN05_Dunf_Balo_h[7]
        ZeroPup = ZeroPup + PresDrop_PN05_Dunf_Balo_h[8]
        Lpacknet[h] = Lpacknet[h] + PresDrop_PN05_Dunf_Balo_h[9]
        # print("Lpacknet[",h,"]: ",Lpacknet[h])

        PresDrop_PN06_Balo_Laud_h = PressureDrop(Param_PN06_Balo_Laud[0], Flux_PN06_Balo_Laud[h],
Param_PN06_Balo_Laud[1], Len_PN06_Balo_Laud, \
                                                 PresDrop_PN05_Dunf_Balo[h][1], properties,
"PN06_Balo_Laud",h, Param_PN06_Balo_Laud[2])
        PresDrop_PN06_Balo_Laud.append(PresDrop_PN06_Balo_Laud_h)
        NoCon = NoCon + PresDrop_PN06_Balo_Laud_h[5]
        ZeroD = ZeroD + PresDrop_PN06_Balo_Laud_h[6]
        LowPress = LowPress + PresDrop_PN06_Balo_Laud_h[7]
        ZeroPup = ZeroPup + PresDrop_PN06_Balo_Laud_h[8]
        Lpacknet[h] = Lpacknet[h] + PresDrop_PN06_Balo_Laud_h[9]
        # print("Lpacknet[",h,"]: ",Lpacknet[h])

        ##--- Demand pipes ---##  NB upstream pressure set to a maximum of 700 kPa

        PresDrop_PD19_Petc_Abdn_h = PressureDrop(Param_PD19_Petc_Abdn[0], Flux_PD19_Petc_Abdn[h],
Param_PD19_Petc_Abdn[1], Len_PD19_Petc_Abdn, \
                                                 min(PresDrop_PN00_Sfeg_Petc[h][1],700), properties,
"PD19_Petc_Abdn",h, Param_PD19_Petc_Abdn[2])
        PresDrop_PD19_Petc_Abdn.append(PresDrop_PD19_Petc_Abdn_h)
```

```
        NoCon = NoCon + PresDrop_PD19_Petc_Abdn_h[5]
        ZeroD = ZeroD + PresDrop_PD19_Petc_Abdn_h[6]
        LowPress = LowPress + PresDrop_PD19_Petc_Abdn_h[7]
        ZeroPup = ZeroPup + PresDrop_PD19_Petc_Abdn_h[8]
        Lpackdem.append(PresDrop_PD19_Petc_Abdn_h[9])
        # print("Lpackdem[",h,"]: ",Lpackdem[h])

        PresDrop_PD20_Petc_Absh_h = PressureDrop(Param_PD20_Petc_Absh[0], Flux_PD20_Petc_Absh[h],
Param_PD20_Petc_Absh[1], Len_PD20_Petc_Absh, \
                                                 min(PresDrop_PN00_Sfeg_Petc[h][1],700), properties,
"PD20_Petc_Absh",h, Param_PD20_Petc_Absh[2])
        PresDrop_PD20_Petc_Absh.append(PresDrop_PD20_Petc_Absh_h)
        NoCon = NoCon + PresDrop_PD20_Petc_Absh_h[5]
        ZeroD = ZeroD + PresDrop_PD20_Petc_Absh_h[6]
        LowPress = LowPress + PresDrop_PD20_Petc_Absh_h[7]
        ZeroPup = ZeroPup + PresDrop_PD20_Petc_Absh_h[8]
        Lpackdem[h] = Lpackdem[h] + PresDrop_PD20_Petc_Absh_h[9]
        # print("Lpackdem[",h,"]: ",Lpackdem[h])

        PresDrop_PD21_Foch_Mory_h = PressureDrop(Param_PD21_Foch_Mory[0], Flux_PD21_Foch_Mory[h],
Param_PD21_Foch_Mory[1], Len_PD21_Foch_Mory, \
                                                 min(PresDrop_PN01_Petc_Foch[h][1],700), properties,
"PD21_Foch_Mory",h, Param_PD21_Foch_Mory[2])
        PresDrop_PD21_Foch_Mory.append(PresDrop_PD21_Foch_Mory_h)
        NoCon = NoCon + PresDrop_PD21_Foch_Mory_h[5]
        ZeroD = ZeroD + PresDrop_PD21_Foch_Mory_h[6]
        LowPress = LowPress + PresDrop_PD21_Foch_Mory_h[7]
        ZeroPup = ZeroPup + PresDrop_PD21_Foch_Mory_h[8]
        Lpackdem[h] = Lpackdem[h] + PresDrop_PD21_Foch_Mory_h[9]
        # print("Lpackdem[",h,"]: ",Lpackdem[h])

        PresDrop_PD22_Foch_High_h = PressureDrop(Param_PD22_Foch_High[0], Flux_PD22_Foch_High[h],
Param_PD22_Foch_High[1], Len_PD22_Foch_High, \
                                                 min(PresDrop_PN01_Petc_Foch[h][1],700), properties,
"PD22_Foch_High",h, Param_PD22_Foch_High[2])
        PresDrop_PD22_Foch_High.append(PresDrop_PD22_Foch_High_h)
        NoCon = NoCon + PresDrop_PD22_Foch_High_h[5]
        ZeroD = ZeroD + PresDrop_PD22_Foch_High_h[6]
        LowPress = LowPress + PresDrop_PD22_Foch_High_h[7]
        ZeroPup = ZeroPup + PresDrop_PD22_Foch_High_h[8]
        Lpackdem[h] = Lpackdem[h] + PresDrop_PD22_Foch_High_h[9]
        # print("Lpackdem[",h,"]: ",Lpackdem[h])

        PresDrop_PD23_Kiri_Angs_h = PressureDrop(Param_PD23_Kiri_Angs[0], Flux_PD23_Kiri_Angs[h],
Param_PD23_Kiri_Angs[1], Len_PD23_Kiri_Angs, \
                                                 min(PresDrop_PN02_Petc_Kiri[h][1],700), properties,
"PD23_Kiri_Angs",h, Param_PD23_Kiri_Angs[2])
        PresDrop_PD23_Kiri_Angs.append(PresDrop_PD23_Kiri_Angs_h)
        NoCon = NoCon + PresDrop_PD23_Kiri_Angs_h[5]
        ZeroD = ZeroD + PresDrop_PD23_Kiri_Angs_h[6]
        LowPress = LowPress + PresDrop_PD23_Kiri_Angs_h[7]
        ZeroPup = ZeroPup + PresDrop_PD23_Kiri_Angs_h[8]
        Lpackdem[h] = Lpackdem[h] + PresDrop_PD23_Kiri_Angs_h[9]
        # print("Lpackdem[",h,"]: ",Lpackdem[h])

        PresDrop_PD24_Glen_Dund_h = PressureDrop(Param_PD24_Glen_Dund[0], Flux_PD24_Glen_Dund[h],
Param_PD24_Glen_Dund[1], Len_PD24_Glen_Dund, \
                                                 min(PresDrop_PN03_Kiri_Glen[h][1],700), properties,
"PD24_Glen_Dund",h, Param_PD24_Glen_Dund[2])
        PresDrop_PD24_Glen_Dund.append(PresDrop_PD24_Glen_Dund_h)
        NoCon = NoCon + PresDrop_PD24_Glen_Dund_h[5]
        ZeroD = ZeroD + PresDrop_PD24_Glen_Dund_h[6]
        LowPress = LowPress + PresDrop_PD24_Glen_Dund_h[7]
        ZeroPup = ZeroPup + PresDrop_PD24_Glen_Dund_h[8]
        Lpackdem[h] = Lpackdem[h] + PresDrop_PD24_Glen_Dund_h[9]
        # print("Lpackdem[",h,"]: ",Lpackdem[h])

        PresDrop_PD25_Glen_Pkin_h = PressureDrop(Param_PD25_Glen_Pkin[0], Flux_PD25_Glen_Pkin[h],
Param_PD25_Glen_Pkin[1], Len_PD25_Glen_Pkin, \
                                                 min(PresDrop_PN03_Kiri_Glen[h][1],700), properties,
"PD25_Glen_Pkin",h, Param_PD25_Glen_Pkin[2])
        PresDrop_PD25_Glen_Pkin.append(PresDrop_PD25_Glen_Pkin_h)
        NoCon = NoCon + PresDrop_PD25_Glen_Pkin_h[5]
        ZeroD = ZeroD + PresDrop_PD25_Glen_Pkin_h[6]
```

```
        LowPress = LowPress + PresDrop_PD25_Glen_Pkin_h[7]
        ZeroPup = ZeroPup + PresDrop_PD25_Glen_Pkin_h[8]
        Lpackdem[h] = Lpackdem[h] + PresDrop_PD25_Glen_Pkin_h[9]
        # print("Lpackdem[",h,"]: ",Lpackdem[h])

        PresDrop_PD26_Dunf_Fife_h = PressureDrop(Param_PD26_Dunf_Fife[0], Flux_PD26_Dunf_Fife[h],
Param_PD26_Dunf_Fife[1], Len_PD26_Dunf_Fife, \
                                                 min(PresDrop_PN04_Glen_Dunf[h][1],700), properties,
"PD26_Dunf_Fife",h, Param_PD26_Dunf_Fife[2])
        PresDrop_PD26_Dunf_Fife.append(PresDrop_PD26_Dunf_Fife_h)
        NoCon = NoCon + PresDrop_PD26_Dunf_Fife_h[5]
        ZeroD = ZeroD + PresDrop_PD26_Dunf_Fife_h[6]
        LowPress = LowPress + PresDrop_PD26_Dunf_Fife_h[7]
        ZeroPup = ZeroPup + PresDrop_PD26_Dunf_Fife_h[8]
        Lpackdem[h] = Lpackdem[h] + PresDrop_PD26_Dunf_Fife_h[9]
        # print("Lpackdem[",h,"]: ",Lpackdem[h])

        PresDrop_PD27_Balo_Wlot_h = PressureDrop(Param_PD27_Balo_Wlot[0], Flux_PD27_Balo_Wlot[h],
Param_PD27_Balo_Wlot[1], Len_PD27_Balo_Wlot, \
                                                 min(PresDrop_PN05_Dunf_Balo[h][1],700), properties,
"PD27_Balo_Wlot",h, Param_PD27_Balo_Wlot[2])
        PresDrop_PD27_Balo_Wlot.append(PresDrop_PD27_Balo_Wlot_h)
        NoCon = NoCon + PresDrop_PD27_Balo_Wlot_h[5]
        ZeroD = ZeroD + PresDrop_PD27_Balo_Wlot_h[6]
        LowPress = LowPress + PresDrop_PD27_Balo_Wlot_h[7]
        ZeroPup = ZeroPup + PresDrop_PD27_Balo_Wlot_h[8]
        Lpackdem[h] = Lpackdem[h] + PresDrop_PD27_Balo_Wlot_h[9]
        # print("Lpackdem[",h,"]: ",Lpackdem[h])

        PresDrop_PD28_Balo_Edin_h = PressureDrop(Param_PD28_Balo_Edin[0], Flux_PD28_Balo_Edin[h],
Param_PD28_Balo_Edin[1], Len_PD28_Balo_Edin, \
                                                 min(PresDrop_PN05_Dunf_Balo[h][1],700), properties,
"PD28_Balo_Edin",h, Param_PD28_Balo_Edin[2])
        PresDrop_PD28_Balo_Edin.append(PresDrop_PD28_Balo_Edin_h)
        NoCon = NoCon + PresDrop_PD28_Balo_Edin_h[5]
        ZeroD = ZeroD + PresDrop_PD28_Balo_Edin_h[6]
        LowPress = LowPress + PresDrop_PD28_Balo_Edin_h[7]
        ZeroPup = ZeroPup + PresDrop_PD28_Balo_Edin_h[8]
        Lpackdem[h] = Lpackdem[h] + PresDrop_PD28_Balo_Edin_h[9]
        # print("Lpackdem[",h,"]: ",Lpackdem[h])

        PresDrop_PD29_Balo_Mlot_h = PressureDrop(Param_PD29_Balo_Mlot[0], Flux_PD29_Balo_Mlot[h],
Param_PD29_Balo_Mlot[1], Len_PD29_Balo_Mlot, \
                                                 min(PresDrop_PN05_Dunf_Balo[h][1],700), properties,
"PD29_Balo_Mlot",h, Param_PD29_Balo_Mlot[2])
        PresDrop_PD29_Balo_Mlot.append(PresDrop_PD29_Balo_Mlot_h)
        NoCon = NoCon + PresDrop_PD29_Balo_Mlot_h[5]
        ZeroD = ZeroD + PresDrop_PD29_Balo_Mlot_h[6]
        LowPress = LowPress + PresDrop_PD29_Balo_Mlot_h[7]
        ZeroPup = ZeroPup + PresDrop_PD29_Balo_Mlot_h[8]
        Lpackdem[h] = Lpackdem[h] + PresDrop_PD29_Balo_Mlot_h[9]
        # print("Lpackdem[",h,"]: ",Lpackdem[h])

        PresDrop_PD30_Laud_Elot_h = PressureDrop(Param_PD30_Laud_Elot[0], Flux_PD30_Laud_Elot[h],
Param_PD30_Laud_Elot[1], Len_PD30_Laud_Elot, \
                                                 min(PresDrop_PN06_Balo_Laud[h][1],700), properties,
"PD30_Laud_Elot",h, Param_PD30_Laud_Elot[2])
        PresDrop_PD30_Laud_Elot.append(PresDrop_PD30_Laud_Elot_h)
        NoCon = NoCon + PresDrop_PD30_Laud_Elot_h[5]
        ZeroD = ZeroD + PresDrop_PD30_Laud_Elot_h[6]
        LowPress = LowPress + PresDrop_PD30_Laud_Elot_h[7]
        ZeroPup = ZeroPup + PresDrop_PD30_Laud_Elot_h[8]
        Lpackdem[h] = Lpackdem[h] + PresDrop_PD30_Laud_Elot_h[9]
        # print("Lpackdem[",h,"]: ",Lpackdem[h])

        PresDrop_PD31_Laud_Sbor_h = PressureDrop(Param_PD31_Laud_Sbor[0], Flux_PD31_Laud_Sbor[h],
Param_PD31_Laud_Sbor[1], Len_PD31_Laud_Sbor, \
                                                 min(PresDrop_PN06_Balo_Laud[h][1],700), properties,
"PD31_Laud_Sbor",h, Param_PD31_Laud_Sbor[2])
        PresDrop_PD31_Laud_Sbor.append(PresDrop_PD31_Laud_Sbor_h)
        NoCon = NoCon + PresDrop_PD31_Laud_Sbor_h[5]
        ZeroD = ZeroD + PresDrop_PD31_Laud_Sbor_h[6]
        LowPress = LowPress + PresDrop_PD31_Laud_Sbor_h[7]
        ZeroPup = ZeroPup + PresDrop_PD31_Laud_Sbor_h[8]
```

```
        Lpackdem[h] = Lpackdem[h] + PresDrop_PD31_Laud_Sbor_h[9]
        # print("Lpackdem[",h,"]: ",Lpackdem[h])

        Lpacktot.append(Lpacknet[h] + Lpackdem[h])

    # Print noconverge and zerodrop error checks
    print("",file=fileout)
    print("Non-converging pipe count: ",NoCon,file=fileout)
    print("Zero drop pipe count: ",ZeroD,file=fileout)
    print("Low pressure pipe count: ",LowPress,file = fileout)
    print("zero pressure upstream: ", ZeroPup, file = fileout)

    # Print Pressure drop output table, network pipes
    print("",file=fileout)
    print("Pressure drop table - network pipes (kPa)", file = fileout)
    print("{:<5} {:<14} {:<14} {:<14} {:<14} {:<14} {:<14} {:<14} ".\
        format("Time","PN00 Sfeg Petc", "PN01 Petc Foch", "PN02 Petc Kiri", "PN03 Kiri Glen",
"PN04 Glen Dunf", "PN05 Dunf Balo", "PN06 Balo Laud", ), file=fileout)
    # print("",file=fileout)
    for h in Hours:
        print("{:<5} {:<14} {:<14} {:<14} {:<14} {:<14} {:<14} {:<14} ".\
            format(h, round(PresDrop_PN00_Sfeg_Petc[h][0]), round(PresDrop_PN01_Petc_Foch[h][0]),
round(PresDrop_PN02_Petc_Kiri[h][0]), round(PresDrop_PN03_Kiri_Glen[h][0]),\
                    round(PresDrop_PN04_Glen_Dunf[h][0]), round(PresDrop_PN05_Dunf_Balo[h][0]),
round(PresDrop_PN06_Balo_Laud[h][0]), ),file=fileout)
    print("",file=fileout)

    # Print Pressure drop output table, demand pipes
    print("Pressure drop table - demand pipes (Pa)", file = fileout)
    print("{:<5} {:<14} {:<14} {:<14} {:<14} {:<14} {:<14} {:<14} {:<14} {:<14} {:<14} ".\
        format("Time", "PD19 Petc Abdn", "PD20 Petc Absh", "PD21 Foch Mory", "PD22 Foch High",
"PD23 Kiri Angs", "PD24 Glen Dund", "PD25 Glen Pkin", "PD26 Dunf Fife", \
                "PD27 Balo Wlot", "PD28 Balo Edin", "PD29 Balo Mlot", "PD30 Laud Elot", "PD31 Laud Sbor"),
file=fileout)
    # print("",file=fileout)

    for h in Hours:
        print("{:<5} {:<14} {:<14} {:<14} {:<14} {:<14} {:<14} {:<14} {:<14} {:<14} {:<14} {:<14}
{:<14} ".\
            format(h, round(PresDrop_PD19_Petc_Abdn[h][0]), round(PresDrop_PD20_Petc_Absh[h][0]),
round(PresDrop_PD21_Foch_Mory[h][0]), round(PresDrop_PD22_Foch_High[h][0]),\
                    round(PresDrop_PD23_Kiri_Angs[h][0]), round(PresDrop_PD24_Glen_Dund[h][0]),
round(PresDrop_PD25_Glen_Pkin[h][0]), round(PresDrop_PD26_Dunf_Fife[h][0]),\
                        round(PresDrop_PD27_Balo_Wlot[h][0]), round(PresDrop_PD28_Balo_Edin[h][0]),
round(PresDrop_PD29_Balo_Mlot[h][0]), round(PresDrop_PD30_Laud_Elot[h][0]),\
                            round(PresDrop_PD31_Laud_Sbor[h][0])),file=fileout)
    print("",file=fileout)

    # Print downstream pressure at nodes
    # Define output arrays
    # Network nodes
    Press_NS00_Sfeg=[]
    Press_NN01_Petc=[]
    Press_NN02_Foch=[]
    Press_NN03_Kiri=[]
    Press_NN04_Glen=[]
    Press_NN05_Dunf=[]
    Press_NN06_Balo=[]
    Press_NN07_Laud=[]
    #Press_NN16_Cold=[]

    # Demand nodes
    Press_ND26_Wlot=[]
    Press_ND27_Edin=[]
    Press_ND28_Mlot=[]
    Press_ND29_Elot=[]
    Press_ND31_Fife=[]
    Press_ND32_Dund=[]
    Press_ND33_Angs=[]
    Press_ND34_Absh=[]
    Press_ND35_Abdn=[]
    Press_ND36_Mory=[]
    Press_ND37_High=[]
    Press_ND39_Pkin=[]
```

```
    Press_ND46_Sbor=[]

    for h in Hours:
        # Append to arrays
        # Network nodes
        Press_NS00_Sfeg.append(InPres_NS00_Sfeg)
        Press_NN01_Petc.append(PresDrop_PN00_Sfeg_Petc[h][1])
        Press_NN02_Foch.append(PresDrop_PN01_Petc_Foch[h][1])
        Press_NN03_Kiri.append(PresDrop_PN02_Petc_Kiri[h][1])
        Press_NN04_Glen.append(PresDrop_PN03_Kiri_Glen[h][1])
        Press_NN05_Dunf.append(PresDrop_PN04_Glen_Dunf[h][1])
        Press_NN06_Balo.append(PresDrop_PN05_Dunf_Balo[h][1])
        Press_NN07_Laud.append(PresDrop_PN06_Balo_Laud[h][1])
        #Press_NN16_Cold.append(PresDrop_PN07_Laud_Cold[h][1])

        # Demand nodes
        Press_ND26_Wlot.append(PresDrop_PD27_Balo_Wlot[h][1])
        Press_ND27_Edin.append(PresDrop_PD28_Balo_Edin[h][1])
        Press_ND28_Mlot.append(PresDrop_PD29_Balo_Mlot[h][1])
        Press_ND29_Elot.append(PresDrop_PD30_Laud_Elot[h][1])
        Press_ND31_Fife.append(PresDrop_PD26_Dunf_Fife[h][1])
        Press_ND32_Dund.append(PresDrop_PD24_Glen_Dund[h][1])
        Press_ND33_Angs.append(PresDrop_PD23_Kiri_Angs[h][1])
        Press_ND34_Absh.append(PresDrop_PD20_Petc_Absh[h][1])
        Press_ND35_Abdn.append(PresDrop_PD19_Petc_Abdn[h][1])
        Press_ND36_Mory.append(PresDrop_PD21_Foch_Mory[h][1])
        Press_ND37_High.append(PresDrop_PD22_Foch_High[h][1])
        Press_ND39_Pkin.append(PresDrop_PD25_Glen_Pkin[h][1])
        Press_ND46_Sbor.append(PresDrop_PD31_Laud_Sbor[h][1])

    print("",file=fileout)
    print("Table of Pressure at each node (kPa)", file=fileout)
    # Print node Pressure table, network nodes
    print("Node Pressure table - network nodes (kPa)", file = fileout)
    print("For natural gas, local transmission network carries pressures from 19bar to 85bar, ie 1900 - 8500
kPa, with a minimum of 7bar.", file = fileout)
    print("Source:  SGN Pipeline Engineering Manager, 18/10/22,  Consider the network pipes and nodes here as
equivalent to LTS.", file = fileout)
    print("{:<5} {:<14} {:<14} {:<14} {:<14} {:<14} {:<14} {:<14} ".\
        format("Time","NS00 Sfeg", "NN01 Petc", "NN02 Foch", "NN03 Kiri", "NN04 Glen", "NN05 Dunf",
"NN06 Balo", "NN07 Laud", ), file=fileout)
    # print("",file=fileout)
    for h in Hours:
        print("{:<5} {:<14} {:<14} {:<14} {:<14} {:<14} {:<14} {:<14} ".\
            format(h,round(Press_NS00_Sfeg[h]), round(Press_NN01_Petc[h]), round(Press_NN02_Foch[h]),
round(Press_NN03_Kiri[h]), round(Press_NN04_Glen[h]), \
                    round(Press_NN05_Dunf[h]), round(Press_NN06_Balo[h]), round(Press_NN07_Laud[h]),
),file=fileout)
    print("",file=fileout)

    ###--- key extract, timestep 7, printed below instead of to file ---###
    print("Pressure at network nodes node, timestep 7 (kPa)")
    print("{:<5} {:<14} {:<14} {:<14} {:<14} {:<14} {:<14} {:<14} ".\
        format("Time","NS00 Sfeg", "NN01 Petc", "NN02 Foch", "NN03 Kiri", "NN04 Glen", "NN05 Dunf",
"NN06 Balo", "NN07 Laud", ))
    print("{:<5} {:<14} {:<14} {:<14} {:<14} {:<14} {:<14} {:<14} ".\
        format("7",round(Press_NS00_Sfeg[7]), round(Press_NN01_Petc[7]), round(Press_NN02_Foch[7]),
round(Press_NN03_Kiri[7]), round(Press_NN04_Glen[7]),\
                round(Press_NN05_Dunf[7]), round(Press_NN06_Balo[7]), round(Press_NN07_Laud[7]), ))

    # Print node Pressure table, demand nodes
    print("Node Pressure table - demand nodes (kPa)", file = fileout)
    print("For natural gas, intermediate pressure pipes carry pressures from 2-7 bar, ie 200-700 kPa.
Source:  SGN Pipeline Engineering Manager, 18/10/22.", file = fileout)
    print("Consider the demand pipes and nodes here as equivalent to IP mains, so a minimum of 200kPa is
required at the demand node." , file = fileout)
    print("{:<5} {:<10} {:<10} {:<10} {:<10} {:<10} {:<10} {:<10} {:<10} {:<10} {:<10}".\
        format("Time", "ND35 Abdn", "ND34 Absh", "ND36 Mory", "ND37 High",
"ND33 Angs",  "ND32 Dund",  "ND39 Pkin", "ND31 Fife", "ND26 Wlot", "ND27 Edin", \
                "ND28 Mlot", "ND29 Elot", "ND46 Sbor"), file=fileout)
    # print("", file=fileout)
    for h in Hours:
        print("{:<5} {:<10} {:<10} {:<10} {:<10} {:<10} {:<10} {:<10} {:<10} {:<10} {:<10} {:<10}
{:<10} ".\
```

```
            format(h, round(Press_ND35_Abdn[h]), round(Press_ND34_Absh[h]), round(Press_ND36_Mory[h]),
round(Press_ND37_High[h]), round(Press_ND33_Angs[h]), \
                    round(Press_ND32_Dund[h]), round(Press_ND39_Pkin[h]), \
                     round(Press_ND31_Fife[h]), round(Press_ND26_Wlot[h]), round(Press_ND27_Edin[h]),
round(Press_ND28_Mlot[h]), round(Press_ND29_Elot[h]), \
                          round(Press_ND46_Sbor[h])),file=fileout)
    print("",file=fileout)

    # Print upstream pressure for demand pipes
    print("Node Pressure table - upstream pressure used for demand nodes (kPa)", file = fileout)
    print("For natural gas, intermediate pressure pipes carry pressures from 2-7 bar, ie 200-700 kPa.
Source:  SGN Pipeline Engineering Manager, 18/10/22.", file = fileout)
    print("Consider the demand pipes and nodes here as equivalent to IP mains." , file = fileout)
    print("{:<5} {:<10} {:<10} {:<10} {:<10} {:<10} {:<10} {:<10} {:<10} {:<10} {:<10} {:<10} {:<10}".\
        format("Time", "ND26_Wlot", "ND27_Edin", "ND28_Mlot", "ND29_Elot", "ND31_Fife", "ND32_Dund",
"ND33_Angs", "ND34_Absh", "ND35_Abdn", "ND36_Mory",\
               "ND37_High", "ND39_Pkin", "ND46_Sbor"), file=fileout)
    # print("", file=fileout)
    for h in Hours:
        print("{:<5} {:<10} {:<10} {:<10} {:<10} {:<10} {:<10} {:<10} {:<10} {:<10} {:<10} {:<10}
{:<10} ".\
            format(h, round(min(Press_NN06_Balo[h],700)), round(min(Press_NN06_Balo[h],700)),
round(min(Press_NN06_Balo[h],700)), round(min(Press_NN07_Laud[h],700)),\
                   round(min(Press_NN05_Dunf[h],700)), round(min(Press_NN04_Glen[h],700)),
round(min(Press_NN03_Kiri[h],700)), round(min(Press_NN01_Petc[h],700)),\
                    round(min(Press_NN01_Petc[h],700)), round(min(Press_NN02_Foch[h],700)),
round(min(Press_NN02_Foch[h],700)), round(min(Press_NN04_Glen[h],700)),\
                          round(min(Press_NN07_Laud[h],700))),file=fileout)
    print("",file=fileout)

    print("",file=fileout)
    print("Table of flow velocities in each pipe (m/s)", file=fileout)
    # Print pipe velocity table, network pipes
    print("Pipe velocity table - network pipes (m/s)", file = fileout)
    print("For natural gas, allowable flow velocity = 40 m/s, though preferably below 20 m/s.", file =
fileout)
    print("Source:  SGN Pipeline Engineering Manager, 18/10/22.", file = fileout)
    print("{:<5} {:<14} {:<14} {:<14} {:<14} {:<14} {:<14} {:<14}  ".\
        format("Time","PN00_Sfeg_Petc", "PN01_Petc_Foch", "PN02_Petc_Kiri", "PN03_Kiri_Glen",
"PN04_Glen_Dunf", "PN05_Dunf_Balo", "PN06_Balo_Laud", ), file=fileout)
    # print("",file=fileout)
    for h in Hours:
        print("{:<5} {:<14} {:<14} {:<14} {:<14} {:<14} {:<14} {:<14} ".\
            format(h,round(PresDrop_PN00_Sfeg_Petc[h][2],3), round(PresDrop_PN01_Petc_Foch[h][2],3),
round(PresDrop_PN02_Petc_Kiri[h][2],3),\
                   round(PresDrop_PN03_Kiri_Glen[h][2],3),
round(PresDrop_PN04_Glen_Dunf[h][2],3),round(PresDrop_PN05_Dunf_Balo[h][2],3),\
                    round(PresDrop_PN06_Balo_Laud[h][2],3)),file=fileout)
    print("",file=fileout)

    print("For natural gas, allowable flow velocity = 40 m/s, though preferably below 20 m/s.", file =
fileout)
    print("Source:  SGN Pipeline Engineering Manager, 18/10/22", file = fileout)
    print("{:<5} {:<15} {:<15} {:<15} {:<15} {:<15} {:<15} {:<15} {:<15} {:<15} {:<15} {:<15} {:<15}
".\
        format("Time","PD19_Petc_Abdn", "PD20_Petc_Absh", "PD21_Foch_Mory", "PD22_Foch_High",
"PD23_Kiri_Angs", "PD24_Glen_Dund", "PD25_Glen_Pkin",\
               "PD26_Dunf_Fife", "PD27_Balo_Wlot", "PD28_Balo_Edin", "PD29_Balo_Mlot", "PD30_Laud_Elot",
"PD31_Laud_Sbor"), file=fileout)
    # print("",file=fileout)
    for h in Hours:
        print("{:<5} {:<15} {:<15} {:<15} {:<15} {:<15} {:<15} {:<15} {:<15} {:<15} {:<15} {:<15}
{:<15} ".\
            format(h, round(PresDrop_PD19_Petc_Abdn[h][2],3), round(PresDrop_PD20_Petc_Absh[h][2],3),
round(PresDrop_PD21_Foch_Mory[h][2],3),\
                   round(PresDrop_PD22_Foch_High[h][2],3), round(PresDrop_PD23_Kiri_Angs[h][2],3),
round(PresDrop_PD24_Glen_Dund[h][2],3),\
                    round(PresDrop_PD25_Glen_Pkin[h][2],3), round(PresDrop_PD26_Dunf_Fife[h][2],3),
round(PresDrop_PD27_Balo_Wlot[h][2],3), \
                          round(PresDrop_PD28_Balo_Edin[h][2],3), round(PresDrop_PD29_Balo_Mlot[h][2],3),
round(PresDrop_PD30_Laud_Elot[h][2],3),\
                               round(PresDrop_PD31_Laud_Sbor[h][2],3)),file=fileout)
    print("",file=fileout)
```

```
    # Print linepack tables
    print("Linepack totals", file = fileout)
    print("{:<5} {:<20} {:<28} {:<20}".\
          format("Time","Linepack, network", "Change in Linepack available", "Upstream Mass flow"),
file=fileout)
    print("{:<5} {:<20} {:<28} {:<20}".\
    format("","(kg)", "(kg)", "(kg/hour)"), file=fileout)

    for h in Hours:
        if h == 0:
            print("{:<5,} {:<20,} {:<28,} {:<20,} ".\
                format(h, round(Lpacknet[h]), round(Lpacknet[0]-Lpacknet[23]),
round(Mflo_PN00_Sfeg_Petc[h])),file=fileout)
        else:
            print("{:<5,} {:<20,} {:<28,} {:<20,}".\
                format(h, round(Lpacknet[h]), round(Lpacknet[h]-Lpacknet[h-1]),
round(Mflo_PN00_Sfeg_Petc[h])),file=fileout)
    print("",file=fileout)

    # Main output table
    print("Combined output table", file = fileout)
    for h in Hours:
        print("", file = fileout)
        print("{:<5} {:<15} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13}
{:<13} {:<13} {:<13} ".\
            format("Time","Pipe", "Diameter", "Length", "Qe (demand)", "gas", "Pup", "Pdown", "Pdrop", "Qm
(achieved)", "V", "Linepack", "noconverge", \
                   "zerodrop", "lowpressure", "NoPup"), file=fileout)
        print("{:<5} {:<15} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13}
{:<13} {:<13} {:<13} ".\
          format("","", "(m)", "(m)", "(kWh/s)", "", "(kPa)", "(kPa)", "(kPa)", "(kg/hr)", "(m/s)", "(kg)",
"", "", "", ""), file=fileout)
        print("Network pipes", file = fileout)

        print("{:<5} {:<15} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13}
{:<13} {:<13} {:<13} ".\
            format(h, "PN00_Sfeg_Petc", Param_PN00_Sfeg_Petc[0],
round(Len_PN00_Sfeg_Petc),round(Flux_PN00_Sfeg_Petc[h],2), gas, round(Press_NS00_Sfeg[h]),
round(Press_NN01_Petc[h]), round(PresDrop_PN00_Sfeg_Petc[h][0]),
round(PresDrop_PN00_Sfeg_Petc[h][10]),round(PresDrop_PN00_Sfeg_Petc[h][2],3),
round(PresDrop_PN00_Sfeg_Petc[h][9]), PresDrop_PN00_Sfeg_Petc[h][5], PresDrop_PN00_Sfeg_Petc[h][6],
PresDrop_PN00_Sfeg_Petc[h][7], PresDrop_PN00_Sfeg_Petc[h][8]), file=fileout)

        print("{:<5} {:<15} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13}
{:<13} {:<13} {:<13} ".\
            format(h, "PN01_Petc_Foch", Param_PN01_Petc_Foch[0],
round(Len_PN01_Petc_Foch),round(Flux_PN01_Petc_Foch[h],2), gas, round(Press_NN01_Petc[h]),
round(Press_NN02_Foch[h]), round(PresDrop_PN01_Petc_Foch[h][0]),
round(PresDrop_PN01_Petc_Foch[h][10]),round(PresDrop_PN01_Petc_Foch[h][2],3),
round(PresDrop_PN01_Petc_Foch[h][9]), PresDrop_PN01_Petc_Foch[h][5], PresDrop_PN01_Petc_Foch[h][6],
PresDrop_PN01_Petc_Foch[h][7], PresDrop_PN01_Petc_Foch[h][8]), file=fileout)

        print("{:<5} {:<15} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13}
{:<13} {:<13} {:<13} ".\
            format(h, "PN02_Petc_Kiri", Param_PN02_Petc_Kiri[0],
round(Len_PN02_Petc_Kiri),round(Flux_PN02_Petc_Kiri[h],2), gas, round(Press_NN01_Petc[h]),
round(Press_NN03_Kiri[h]), round(PresDrop_PN02_Petc_Kiri[h][0]),
round(PresDrop_PN02_Petc_Kiri[h][10]),round(PresDrop_PN02_Petc_Kiri[h][2],3),
round(PresDrop_PN02_Petc_Kiri[h][9]), PresDrop_PN02_Petc_Kiri[h][5], PresDrop_PN02_Petc_Kiri[h][6],
PresDrop_PN02_Petc_Kiri[h][7], PresDrop_PN02_Petc_Kiri[h][8]), file=fileout)

        print("{:<5} {:<15} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13}
{:<13} {:<13} {:<13} ".\
            format(h, "PN03_Kiri_Glen", Param_PN03_Kiri_Glen[0],
round(Len_PN03_Kiri_Glen),round(Flux_PN03_Kiri_Glen[h],2), gas, round(Press_NN03_Kiri[h]),
round(Press_NN04_Glen[h]), round(PresDrop_PN03_Kiri_Glen[h][0]),
round(PresDrop_PN03_Kiri_Glen[h][10]),round(PresDrop_PN03_Kiri_Glen[h][2],3),
round(PresDrop_PN03_Kiri_Glen[h][9]), PresDrop_PN03_Kiri_Glen[h][5], PresDrop_PN03_Kiri_Glen[h][6],
PresDrop_PN03_Kiri_Glen[h][7], PresDrop_PN03_Kiri_Glen[h][8]), file=fileout)

        print("{:<5} {:<15} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13}
{:<13} {:<13} {:<13} ".\
            format(h, "PN04_Glen_Dunf", Param_PN04_Glen_Dunf[0],
round(Len_PN04_Glen_Dunf),round(Flux_PN04_Glen_Dunf[h],2), gas, round(Press_NN04_Glen[h]),
```

```
round(Press_NN05_Dunf[h]), round(PresDrop_PN04_Glen_Dunf[h][0]),
round(PresDrop_PN04_Glen_Dunf[h][10]),round(PresDrop_PN04_Glen_Dunf[h][2],3),
round(PresDrop_PN04_Glen_Dunf[h][9]), PresDrop_PN04_Glen_Dunf[h][5], PresDrop_PN04_Glen_Dunf[h][6],
PresDrop_PN04_Glen_Dunf[h][7], PresDrop_PN04_Glen_Dunf[h][8]), file=fileout)

        print("{:<5} {:<15} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13}
{:<13} {:<13} {:<13} ".\
            format(h, "PN05_Dunf_Balo", Param_PN05_Dunf_Balo[0],
round(Len_PN05_Dunf_Balo),round(Flux_PN05_Dunf_Balo[h],2), gas, round(Press_NN05_Dunf[h]),
round(Press_NN06_Balo[h]), round(PresDrop_PN05_Dunf_Balo[h][0]),
round(PresDrop_PN05_Dunf_Balo[h][10]),round(PresDrop_PN05_Dunf_Balo[h][2],3),
round(PresDrop_PN05_Dunf_Balo[h][9]), PresDrop_PN05_Dunf_Balo[h][5], PresDrop_PN05_Dunf_Balo[h][6],
PresDrop_PN05_Dunf_Balo[h][7], PresDrop_PN05_Dunf_Balo[h][8]), file=fileout)

        print("{:<5} {:<15} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13}
{:<13} {:<13} {:<13} ".\
            format(h, "PN06_Balo_Laud", Param_PN06_Balo_Laud[0],
round(Len_PN06_Balo_Laud),round(Flux_PN06_Balo_Laud[h],2), gas, round(Press_NN06_Balo[h]),
round(Press_NN07_Laud[h]), round(PresDrop_PN06_Balo_Laud[h][0]),
round(PresDrop_PN06_Balo_Laud[h][10]),round(PresDrop_PN06_Balo_Laud[h][2],3),
round(PresDrop_PN06_Balo_Laud[h][9]), PresDrop_PN06_Balo_Laud[h][5], PresDrop_PN06_Balo_Laud[h][6],
PresDrop_PN06_Balo_Laud[h][7], PresDrop_PN06_Balo_Laud[h][8]), file=fileout)

        print("Demand pipes", file = fileout)
        print("{:<5} {:<15} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13}
{:<13} {:<13} {:<13} ".\
            format(h, "PD19_Petc_Abdn", Param_PD19_Petc_Abdn[0], round(Len_PD19_Petc_Abdn),
round(Flux_PD19_Petc_Abdn[h],2), gas, round(min(Press_NN01_Petc[h],700)), round(Press_ND35_Abdn[h]),
round(PresDrop_PD19_Petc_Abdn[h][0]), round(PresDrop_PD19_Petc_Abdn[h][10]),
round(PresDrop_PD19_Petc_Abdn[h][2],3), round(PresDrop_PD19_Petc_Abdn[h][9]), PresDrop_PD19_Petc_Abdn[h][5],
PresDrop_PD19_Petc_Abdn[h][6], PresDrop_PD19_Petc_Abdn[h][7], PresDrop_PD19_Petc_Abdn[h][8]),
file=fileout)

        print("{:<5} {:<15} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13}
{:<13} {:<13} {:<13} ".\
            format(h, "PD20_Petc_Absh", Param_PD20_Petc_Absh[0], round(Len_PD20_Petc_Absh),
round(Flux_PD20_Petc_Absh[h],2), gas, round(min(Press_NN01_Petc[h],700)), round(Press_ND34_Absh[h]),
round(PresDrop_PD20_Petc_Absh[h][0]), round(PresDrop_PD20_Petc_Absh[h][10]),
round(PresDrop_PD20_Petc_Absh[h][2],3), round(PresDrop_PD20_Petc_Absh[h][9]), PresDrop_PD20_Petc_Absh[h][5],
PresDrop_PD20_Petc_Absh[h][6], PresDrop_PD20_Petc_Absh[h][7], PresDrop_PD20_Petc_Absh[h][8]),
file=fileout)

        print("{:<5} {:<15} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13}
{:<13} {:<13} {:<13} ".\
            format(h, "PD21_Foch_Mory", Param_PD21_Foch_Mory[0], round(Len_PD21_Foch_Mory),
round(Flux_PD21_Foch_Mory[h],2), gas, round(min(Press_NN02_Foch[h],700)), round(Press_ND36_Mory[h]),
round(PresDrop_PD21_Foch_Mory[h][0]), round(PresDrop_PD21_Foch_Mory[h][10]),
round(PresDrop_PD21_Foch_Mory[h][2],3), round(PresDrop_PD21_Foch_Mory[h][9]), PresDrop_PD21_Foch_Mory[h][5],
PresDrop_PD21_Foch_Mory[h][6], PresDrop_PD21_Foch_Mory[h][7], PresDrop_PD21_Foch_Mory[h][8]),
file=fileout)

        print("{:<5} {:<15} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13}
{:<13} {:<13} {:<13} ".\
            format(h, "PD22_Foch_High", Param_PD22_Foch_High[0], round(Len_PD22_Foch_High),
round(Flux_PD22_Foch_High[h],2), gas, round(min(Press_NN02_Foch[h],700)), round(Press_ND37_High[h]),
round(PresDrop_PD22_Foch_High[h][0]), round(PresDrop_PD22_Foch_High[h][10]),
round(PresDrop_PD22_Foch_High[h][2],3), round(PresDrop_PD22_Foch_High[h][9]), PresDrop_PD22_Foch_High[h][5],
PresDrop_PD22_Foch_High[h][6], PresDrop_PD22_Foch_High[h][7], PresDrop_PD22_Foch_High[h][8]),
file=fileout)

        print("{:<5} {:<15} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13}
{:<13} {:<13} {:<13} ".\
            format(h, "PD23_Kiri_Angs", Param_PD23_Kiri_Angs[0], round(Len_PD23_Kiri_Angs),
round(Flux_PD23_Kiri_Angs[h],2), gas, round(min(Press_NN03_Kiri[h],700)), round(Press_ND33_Angs[h]),
round(PresDrop_PD23_Kiri_Angs[h][0]), round(PresDrop_PD23_Kiri_Angs[h][10]),
round(PresDrop_PD23_Kiri_Angs[h][2],3), round(PresDrop_PD23_Kiri_Angs[h][9]), PresDrop_PD23_Kiri_Angs[h][5],
PresDrop_PD23_Kiri_Angs[h][6], PresDrop_PD23_Kiri_Angs[h][7], PresDrop_PD23_Kiri_Angs[h][8]),
file=fileout)

        print("{:<5} {:<15} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13}
{:<13} {:<13} {:<13} ".\
            format(h, "PD24_Glen_Dund", Param_PD24_Glen_Dund[0], round(Len_PD24_Glen_Dund),
round(Flux_PD24_Glen_Dund[h],2), gas, round(min(Press_NN04_Glen[h],700)), round(Press_ND32_Dund[h]),
round(PresDrop_PD24_Glen_Dund[h][0]), round(PresDrop_PD24_Glen_Dund[h][10]),
```

```
round(PresDrop_PD24_Glen_Dund[h][2],3), round(PresDrop_PD24_Glen_Dund[h][9]), PresDrop_PD24_Glen_Dund[h][5],
PresDrop_PD24_Glen_Dund[h][6], PresDrop_PD24_Glen_Dund[h][7], PresDrop_PD24_Glen_Dund[h][8]),
file=fileout)

        print("{:<5} {:<15} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13}
{:<13} {:<13} ".\
            format(h, "PD25_Glen_Pkin", Param_PD25_Glen_Pkin[0], round(Len_PD25_Glen_Pkin),
round(Flux_PD25_Glen_Pkin[h],2), gas, round(min(Press_NN04_Glen[h],700)), round(Press_ND39_Pkin[h]),
round(PresDrop_PD25_Glen_Pkin[h][0]), round(PresDrop_PD25_Glen_Pkin[h][10]),
round(PresDrop_PD25_Glen_Pkin[h][2],3), round(PresDrop_PD25_Glen_Pkin[h][9]), PresDrop_PD25_Glen_Pkin[h][5],
PresDrop_PD25_Glen_Pkin[h][6], PresDrop_PD25_Glen_Pkin[h][7], PresDrop_PD25_Glen_Pkin[h][8]),
file=fileout)

        print("{:<5} {:<15} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13}
{:<13} {:<13} ".\
            format(h, "PD26_Dunf_Fife", Param_PD26_Dunf_Fife[0], round(Len_PD26_Dunf_Fife),
round(Flux_PD26_Dunf_Fife[h],2), gas, round(min(Press_NN05_Dunf[h],700)), round(Press_ND31_Fife[h]),
round(PresDrop_PD26_Dunf_Fife[h][0]), round(PresDrop_PD26_Dunf_Fife[h][10]),
round(PresDrop_PD26_Dunf_Fife[h][2],3), round(PresDrop_PD26_Dunf_Fife[h][9]), PresDrop_PD26_Dunf_Fife[h][5],
PresDrop_PD26_Dunf_Fife[h][6], PresDrop_PD26_Dunf_Fife[h][7], PresDrop_PD26_Dunf_Fife[h][8]),
file=fileout)

        print("{:<5} {:<15} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13}
{:<13} {:<13} ".\
            format(h, "PD27_Balo_Wlot", Param_PD27_Balo_Wlot[0], round(Len_PD27_Balo_Wlot),
round(Flux_PD27_Balo_Wlot[h],2), gas, round(min(Press_NN06_Balo[h],700)), round(Press_ND26_Wlot[h]),
round(PresDrop_PD27_Balo_Wlot[h][0]), round(PresDrop_PD27_Balo_Wlot[h][10]),
round(PresDrop_PD27_Balo_Wlot[h][2],3), round(PresDrop_PD27_Balo_Wlot[h][9]), PresDrop_PD27_Balo_Wlot[h][5],
PresDrop_PD27_Balo_Wlot[h][6], PresDrop_PD27_Balo_Wlot[h][7], PresDrop_PD27_Balo_Wlot[h][8]),
file=fileout)

        print("{:<5} {:<15} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13}
{:<13} {:<13} ".\
            format(h, "PD28_Balo_Edin", Param_PD28_Balo_Edin[0], round(Len_PD28_Balo_Edin),
round(Flux_PD28_Balo_Edin[h],2), gas, round(min(Press_NN06_Balo[h],700)), round(Press_ND27_Edin[h]),
round(PresDrop_PD28_Balo_Edin[h][0]), round(PresDrop_PD28_Balo_Edin[h][10]),
round(PresDrop_PD28_Balo_Edin[h][2],3), round(PresDrop_PD28_Balo_Edin[h][9]), PresDrop_PD28_Balo_Edin[h][5],
PresDrop_PD28_Balo_Edin[h][6], PresDrop_PD28_Balo_Edin[h][7], PresDrop_PD28_Balo_Edin[h][8]),
file=fileout)

        print("{:<5} {:<15} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13}
{:<13} {:<13} ".\
            format(h, "PD29_Balo_Mlot", Param_PD29_Balo_Mlot[0], round(Len_PD29_Balo_Mlot),
round(Flux_PD29_Balo_Mlot[h],2), gas, round(min(Press_NN06_Balo[h],700)), round(Press_ND28_Mlot[h]),
round(PresDrop_PD29_Balo_Mlot[h][0]), round(PresDrop_PD29_Balo_Mlot[h][10]),
round(PresDrop_PD29_Balo_Mlot[h][2],3), round(PresDrop_PD29_Balo_Mlot[h][9]), PresDrop_PD29_Balo_Mlot[h][5],
PresDrop_PD29_Balo_Mlot[h][6], PresDrop_PD29_Balo_Mlot[h][7], PresDrop_PD29_Balo_Mlot[h][8]),
file=fileout)

        print("{:<5} {:<15} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13}
{:<13} {:<13} ".\
            format(h, "PD30_Laud_Elot", Param_PD30_Laud_Elot[0], round(Len_PD30_Laud_Elot),
round(Flux_PD30_Laud_Elot[h],2), gas, round(min(Press_NN07_Laud[h],700)), round(Press_ND29_Elot[h]),
round(PresDrop_PD30_Laud_Elot[h][0]), round(PresDrop_PD30_Laud_Elot[h][10]),
round(PresDrop_PD30_Laud_Elot[h][2],3), round(PresDrop_PD30_Laud_Elot[h][9]), PresDrop_PD30_Laud_Elot[h][5],
PresDrop_PD30_Laud_Elot[h][6], PresDrop_PD30_Laud_Elot[h][7], PresDrop_PD30_Laud_Elot[h][8]),
file=fileout)

        print("{:<5} {:<15} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13} {:<13}
{:<13} {:<13} ".\
            format(h, "PD31_Laud_Sbor", Param_PD31_Laud_Sbor[0], round(Len_PD31_Laud_Sbor),
round(Flux_PD31_Laud_Sbor[h],2), gas, round(min(Press_NN07_Laud[h],700)), round(Press_ND46_Sbor[h]),
round(PresDrop_PD31_Laud_Sbor[h][0]), round(PresDrop_PD31_Laud_Sbor[h][10]),
round(PresDrop_PD31_Laud_Sbor[h][2],3), round(PresDrop_PD31_Laud_Sbor[h][9]), PresDrop_PD31_Laud_Sbor[h][5],
PresDrop_PD31_Laud_Sbor[h][6], PresDrop_PD31_Laud_Sbor[h][7], PresDrop_PD31_Laud_Sbor[h][8]),
file=fileout)

    # Print error checks
    print("Non-converging pipe count: ",NoCon,file=fileout)
    print("Zero drop pipe count: ",ZeroD,file=fileout)
    print("Low pressure pipe count: ",LowPress,file = fileout)
    print("zero pressure upstream: ", ZeroPup, file = fileout)

    print("",file=fileout)
```

```
    dt = datetime.now().strftime("%Y-%m-%d %H:%M:%S")
    print("Finished: ", dt, file=fileout)

    fileout.close()
print("finished at ", dt)
print()
```