```
# Python Algorithm

# CALCULATING PRESSURE DROP
# condition: RADIAL, TRANSIENT flow, and SLIGHTLY
COMPRESSIBLE fluid
# Craft & Hawkins, 1959

# initialize Python library
import math
import matplotlib.pyplot as plt
from scipy.special import expi as expi
import numpy as np

# initialize empty array
p_drop = []
x = []
E = []

t = []
r = []

# initialize value for pressure drop calculation
pi = float(2500)        # initial pressure (psia)
q = float(300)  # flow rate (STB/day)
B = float(1.32) # formation volume factor (bbl/STB)
mu = float(0.44)        # viscosity (cp (centipoise))
k = float(25)           # permeability (mD)
h = float(45)           # depth (feet)
c = float(18E-6)        # isothermal compressibility (1/
psi)
phi = float(0.16)       # porosity (pore/bulk volume
ratio)

def x_func(phi, mu, c, r, k, t):
    ”””x function for mathematical exponential
integral function”””
    return round(float(phi * mu * c * r**2 / (0.00105
* k * t * 24)), 10)

def PressureDrop(pi, q, mu, B, k, h, E):
    ”””pressure drop function”””
    return round(pi - (70.6 * q * mu * B / (k * h) *
-1 * E), 10)

# attempt inputs
nt = int(input('masukkan banyak t: \n'))

for i in range(0, nt):
    t.append(float(input('masukkan nilai t ke-' +
str(i + 1) + ': \n')))

nr = int(input('masukkan banyak r: \n'))

for i in range(0, nr):
    r.append(float(input('masukkan nilai r ke-' +
str(i + 1) + ': \n')))

# filling data
for i in range(0, nt):
    # init empty list for pressure drop and x for
integral function
    p_drop.append([])
    x.append([])
    E.append([])

    # filling data process
    for j in range(0, nr):
        x[i].append(x_func(phi, mu, c, r[j], k,
t[i]))
        E[i].append(expi(-1 * x[i][j]))
        p_drop[i].append(PressureDrop(pi, q, mu,
B, k, h, E[i][j]))
        # print(t[i], r[j], x[i][j], E[i][j],
p_drop[i][j])

# save data to csv format
np.savetxt(”x-value.csv”, np.row_stack(x),
delimiter=”,”, fmt='%s')
np.savetxt(”E-value.csv”, np.row_stack(E),
delimiter=”,”, fmt='%s')
np.savetxt(”pressure_drop.csv”, np.row_stack(p_drop),
delimiter=”,”, fmt='%s')

# plotting
fig = plt.figure()
ax1 = fig.add_subplot(1, 1, 1)

# normal plot
for i in range(0, nt):
    ax1.scatter(r, p_drop[i], label=”t = ” +
str(t[i]) + ” days”)

plt.title('PRESURE DROP VALUE\nTRANSIENT RADIAL FLOW,
INCOMPRESSIBLE FLUID\nNORMAL CARTESIAN SCALE')
plt.xlabel('radius (ft)')
plt.ylabel('pressure (psia)')

ax1.legend()
plt.savefig('normal.png')


# semilog plot
fig = plt.figure()
ax2 = fig.add_subplot(1, 1, 1)

for i in range(0, nt):
    ax2.scatter(r, p_drop[i], label=”t = ” +
str(t[i]) + ” days”)

ax2.set_xscale('log')

plt.title('PRESURE DROP VALUE\nTRANSIENT RADIAL FLOW,
INCOMPRESSIBLE FLUID\nNORMAL CARTESIAN SCALE')
plt.xlabel('radius (ft)')
plt.ylabel('pressure (psia)')

ax2.legend()
plt.savefig('semilog.png')
```