```
%reset
```

```
Once deleted, variables cannot be recovered. Proceed (y/[n])? y
```

PART 1 *: Display API*

```
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sus
import random as rand
```

```
class UAV :

    def __init__(self, pId, pFreq) :
        pass
```

```
arr = np.array([[1,2,3,4], [5,6,7,8], [9,10,11,12]])
arr
```

```
array([[ 1,  2,  3,  4],
       [ 5,  6,  7,  8],
       [ 9, 10, 11, 12]])
```

```
np.delete(arr, 2, 0)
```

```
array([[1, 2, 3, 4],
       [5, 6, 7, 8]])
```

```
arr
```

```
array([[ 1,  2,  3,  4],
       [ 5,  6,  7,  8],
       [ 9, 10, 11, 12]])
```

```
np.delete(np.delete(arr,3, 1), 0, 0)
```

```
array([[ 5,  6,  7],
       [ 9, 10, 11]])
```

variable | type | reason d[i][j] |float |distance bw i and j f1 |float |frequency of drone in LoS f2 |float |frequency of drone in NLoS pl[i][j] |float |path loss pl=20log(d*f) p |float |transmission power g |float |gain c[i][j] |float |signal strength

```
"""
g -> channel gain
p -> trainsmission power
```

```
"""
```

```
     '\ng -> channel gain\np -> trainsmission power\n\n'
```

```
# C = []
# C[i][j] = g[i][j] * np.log( 1 + ( p[j] ) / 10**(P*L[i][j]) )


class UAV :

    UAV_ID = None
    x = 0
    y = 0

    def __init__(self, px, py) :

        self.x = px
        self.y = py


class Swarm :

    UAV_matrix = []
    SWARM_ID = [0,0]
    hostUAV = None
    lastNum = 0

    def __init__( self, host ) :

        self.hostUAV = host

    def AddUAV( self, UAV ) :

        if( len(self.UAV_matrix) == self.lastNum // 10 ) :
            self.UAV_matrix.append( [0 for i in range(0,10)] )

        UAV.UAV_ID = [ self.lastNum // 10, self.lastNum % 10]
        # print(UAV.UAV_ID)
        self.UAV_matrix[ self.lastNum // 10 ][self.lastNum % 10] = UAV
        self.lastNum += 1

    def GetUAV( self, UAV_ID ) : return self.SWARM_ID[ UAV_ID // 10 ][UAV_ID % 10]

    def GetAllUAVs( self ) :

        uavs = []
        for i in self.UAV_matrix :
            for j in i :

                if( j != 0 ) : uavs.append( j )

        # print( uavs )
        return uavs

class Environment :

    swarm_matrix = []
    lastNum = 0
    RESOLUTION = []
    space = []

    def ClearSpace ( self, x_res, y_res ) :

        self.space = [ [ "_" for i in range( 0, x_res ) ] for j in range( 0, y_res ) ]

    def __init__(self, x_res, y_res):
        self.RESOLUTION = [ x_res, y_res ]
        self.ClearSpace( x_res , y_res )

    def AddSwarm( self, swarm ) :

        if( len(self.swarm_matrix) == self.lastNum // 10 ) :
            self.swarm_matrix.append( [0 for i in range(0,10)] )

        swarm.SWARM_ID = [ self.lastNum // 10, self.lastNum % 10]
        self.swarm_matrix[ self.lastNum // 10 ][self.lastNum % 10] = swarm
        self.lastNum += 1

    def GetSwarm( self, SWARM_ID ) : return self.swarm_matrix[ SWARM_ID // 10 ][SWARM_ID %

    def Render( self ) :

        for axis in self.space : print( axis )

    def Get( self ) :

        print( self.swarm_matrix )
        self.swarm_matrix[-1][-1] = "hello"

        print( self.swarm_matrix )

    def SetObject( self, swarm_id, uav_id ) :
        swarm = self.GetSwarm( swarm_id )
        uav = swarm.GetUAV( uav_id )
        self.space[ uav.x ][ uav.y ] = chr(ord('a') + self.swarm_matrix.index( swarm ))

    def GetAllSwarms( self ) :

        swarms = []
        for i in self.swarm_matrix :
            for j in i :

                if( j != 0 ) : swarms.append( j )

        return swarms
    def GetAllUAVs( self ) :

        drones = []
        # print(self.GetAllSwarms())
        for swarm in self.GetAllSwarms() :
            for uav in swarm.GetAllUAVs() :
                drones.append( uav )

        return drones
        # O(n)
        # print( self.swarm_matrix )
        # for swarm_row in self.swarm_matrix :
        #     for swarm in swarm_row :
        #         for uav_row in swarm :
        #             for uav in uav_row :
        #                 print( uav.UAV_ID )

    def Sync( self ) :

        drones = []
        swarm_count = 0
        # print(self.GetAllSwarms())
        for swarm in self.GetAllSwarms() :
            for uav in swarm.GetAllUAVs() :

                self.space[ uav.x ][ uav.y ] = chr(65 + swarm_count)
                print( f"Added drone S_{swarm_count}{uav.UAV_ID}{self.space[ uav.x ][ uav.

            swarm_count += 1


world = Environment( 32, 32 )
world.Render()
```

```
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
```

```
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_',
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_',
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_',
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_',
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_',
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_',
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_',
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_',
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_',
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_',
```

```
swarm1 = None
swarm2 = None
swarm3 = None


uav_01 = UAV(3,5)
uav_02 = UAV(3,10)
uav_03 = UAV(7,9 )
uav_04 = UAV(7,4 )

swarm1 = Swarm( uav_01 )

swarm1.AddUAV( uav_01 )
swarm1.AddUAV( uav_02 )
swarm1.AddUAV( uav_03 )
swarm1.AddUAV( uav_04 )

tmp1 = swarm1


world.AddSwarm( tmp1 )


[ f"({dr.x} {dr.y}) " for dr in swarm1.GetAllUAVs() ]
```

```
['(3 5) ', '(3 10) ', '(7 9) ', '(7 4) ']
```

```
world.Sync()
world.Render()
```

```
Added drone S_0[0, 0]A at (3 5)
Added drone S_0[0, 1]A at (3 10)
Added drone S_0[0, 2]A at (7 9)
Added drone S_0[0, 3]A at (7 4)
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_',
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_',
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_',
['_', '_', '_', '_', '_', 'A', '_', '_', '_', '_', 'A', '_', '_', '_', '_', '_', '_',
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_',
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_',
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_',
['_', '_', '_', '_', 'A', '_', '_', '_', '_', 'A', '_', '_', '_', '_', '_', '_', '_',
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_',
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_',
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_',
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
```

```
uav_11 = UAV(5,15)
uav_12 = UAV(5,20)
uav_13 = UAV(9,19)
uav_14 = UAV(9,14)

swarm2 = Swarm( uav_11 )

swarm2.AddUAV( uav_11 )
swarm2.AddUAV( uav_12 )
swarm2.AddUAV( uav_13 )
swarm2.AddUAV( uav_14 )

tmp2 = swarm2
```

```
[ f"({dr.x} {dr.y}) " for dr in swarm2.GetAllUAVs() ]
```

```
['(5 15) ', '(5 20) ', '(9 19) ', '(9 14) ']
```

```
[ f"({dr.x} {dr.y}) " for dr in swarm1.GetAllUAVs() ]
```

```
['(5 15) ', '(5 20) ', '(9 19) ', '(9 14) ']
```

```
world.AddSwarm( tmp2 )
```

```
world.Sync()
world.Render()
```

```
Added drone S_0[0, 0]A at (5 15)
Added drone S_0[0, 1]A at (5 20)
Added drone S_0[0, 2]A at (9 19)
Added drone S_0[0, 3]A at (9 14)
Added drone S_1[0, 0]B at (5 15)
Added drone S_1[0, 1]B at (5 20)
Added drone S_1[0, 2]B at (9 19)
Added drone S_1[0, 3]B at (9 14)
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_',
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_',
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_',
['_', '_', '_', '_', '_', 'A', '_', '_', '_', '_', 'A', '_', '_', '_', '_', '_', '_',
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_',
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', 'B', '_',
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_',
['_', '_', '_', '_', 'A', '_', '_', '_', '_', 'A', '_', '_', '_', '_', '_', '_', '_',
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_',
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', 'B', '_', '_',
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_',
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_',
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_',
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_',
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_',
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_',
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_',
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_',
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_',
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_',
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_',
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_',
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_',
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_',
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_',
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_',
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_',
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_',
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_',
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_',
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_',
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_',
```

```
uav_21 = UAV(14,10)
uav_22 = UAV(14,15)
uav_23 = UAV(18,12)
uav_24 = UAV(18,8 )

swarm3 = Swarm( uav_21 )

swarm3.AddUAV( uav_21 )
swarm3.AddUAV( uav_22 )
swarm3.AddUAV( uav_23 )
swarm3.AddUAV( uav_24 )

tmp3 = swarm3


[ f"({dr.x} {dr.y}) " for dr in swarm1.GetAllUAVs() ]
```

```
['(14 10) ', '(14 15) ', '(18 12) ', '(18 8) ']
```

```
world.AddSwarm( tmp3 )
```

```
world.Sync()
```

```
Added drone S_0[0, 0]A at (14 10)
Added drone S_0[0, 1]A at (14 15)
Added drone S_0[0, 2]A at (18 12)
Added drone S_0[0, 3]A at (18 8)
Added drone S_1[0, 0]B at (14 10)
Added drone S_1[0, 1]B at (14 15)
Added drone S_1[0, 2]B at (18 12)
Added drone S_1[0, 3]B at (18 8)
Added drone S_2[0, 0]C at (14 10)
Added drone S_2[0, 1]C at (14 15)
Added drone S_2[0, 2]C at (18 12)
Added drone S_2[0, 3]C at (18 8)
```

```
world.Render()
```

```
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', 'A', '_', '_', '_', '_', 'A', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', 'B', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', 'A', '_', '_', '_', '_', 'A', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', 'B', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', 'C', '_', '_', '_', '_', 'C', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', 'C', '_', '_', '_', 'C', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
```

```
swarm1.UAV_matrix
```

```
[[<__main__.UAV at 0x7fdcabc8a400>,
  <__main__.UAV at 0x7fdcabc8a460>,
       <__main__.UAV at 0x7fdcabc8a310>,
       <__main__.UAV at 0x7fdcabc8a3d0>,
       0,
       0,
       0,
       0,
       0,
       0]]
```

```
world.AddSwarm( swarm1 )
```

```
[ f"({dr.x} {dr.y}) " for dr in world.GetAllUAVs() ]
```

```
     ['(14 10) ',
      '(14 15) ',
      '(18 12) ',
      '(18 8) ',
      '(14 10) ',
      '(14 15) ',
      '(18 12) ',
      '(18 8) ',
      '(14 10) ',
      '(14 15) ',
      '(18 12) ',
      '(18 8) ',
      '(14 10) ',
      '(14 15) ',
      '(18 12) ',
      '(18 8) ']
```

```
world.Sync()
```

```
     Added drone S_0[0, 0]A at (14 10)
     Added drone S_0[0, 1]A at (14 15)
     Added drone S_0[0, 2]A at (18 12)
     Added drone S_0[0, 3]A at (18 8)
     Added drone S_1[0, 0]B at (14 10)
     Added drone S_1[0, 1]B at (14 15)
     Added drone S_1[0, 2]B at (18 12)
     Added drone S_1[0, 3]B at (18 8)
     Added drone S_2[0, 0]C at (14 10)
     Added drone S_2[0, 1]C at (14 15)
     Added drone S_2[0, 2]C at (18 12)
     Added drone S_2[0, 3]C at (18 8)
     Added drone S_3[0, 0]D at (14 10)
     Added drone S_3[0, 1]D at (14 15)
     Added drone S_3[0, 2]D at (18 12)
     Added drone S_3[0, 3]D at (18 8)
```

```
world.Render()
```

```
     ['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
     ['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
     ['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
     ['_', '_', '_', '_', '_', 'A', '_', '_', '_', '_', 'A', '_', '_', '_', '_', '_', '_'
     ['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
```

```
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', 'B', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', 'A', '_', '_', '_', '_', 'A', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', 'B', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', 'D', '_', '_', '_', '_', 'D', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', 'D', '_', '_', '_', 'D', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
```

```
swarm1.SWARM_ID
```

```
[0, 3]
```

```
world.GetAllSwarms()
```

```
[<__main__.Swarm at 0x7fdcabc699a0>,
 <__main__.Swarm at 0x7fdcabc73370>,
 <__main__.Swarm at 0x7fdcabc8a340>,
 <__main__.Swarm at 0x7fdcabc699a0>]
```

```
world.GetAllUAVs()[1].UAV_ID
```

```
[0, 1]
```

```
swarm1.lastNum
```

```
4
```

```
for x in swarm1.UAV_matrix :
    for uav in x :
        if( uav != 0 ) : print( uav.UAV_ID )
```

```
[0, 0]
[0, 1]
```

```
    [0, 2]
    [0, 3]
```

```
swarm1.UAV_matrix
```

```
    [[<__main__.UAV at 0x7fdcabc8a400>,
      <__main__.UAV at 0x7fdcabc8a460>,
      <__main__.UAV at 0x7fdcabc8a310>,
      <__main__.UAV at 0x7fdcabc8a3d0>,
      0,
      0,
      0,
      0,
      0,
      0]]
```

```
swarm1.UAV_matrix
```

```
    [[<__main__.UAV at 0x7fdcabc8a400>,
      <__main__.UAV at 0x7fdcabc8a460>,
      <__main__.UAV at 0x7fdcabc8a310>,
      <__main__.UAV at 0x7fdcabc8a3d0>,
      0,
      0,
      0,
      0,
      0,
      0]]
```

```
world.Render()
```

```
    ['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_',
    ['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_',
    ['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_',
    ['_', '_', '_', '_', '_', 'A', '_', '_', '_', '_', 'A', '_', '_', '_', '_', '_', '_',
    ['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_',
    ['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', 'B', '_',
    ['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_',
    ['_', '_', '_', '_', 'A', '_', '_', '_', '_', 'A', '_', '_', '_', '_', '_', '_', '_',
    ['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_',
    ['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', 'B', '_', '_',
    ['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_',
    ['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_',
    ['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_',
    ['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_',
    ['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', 'D', '_', '_', '_', '_', 'D', '_',
    ['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_',
    ['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_',
    ['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_',
    ['_', '_', '_', '_', '_', '_', '_', '_', 'D', '_', '_', '_', 'D', '_', '_', '_', '_',
    ['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_',
    ['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_',
    ['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_',
    ['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_',
    ['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_',
    ['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_',
    ['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_',
    ['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_',
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
['_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_'
```

PART 2 : *Formulating Maths*

Mini API Manual

swarm1, swarm2, swarm3, swarm4 (*Swarm*)

world (*Environment*)

`world.GetAllUAVs()` <- Gets all UAVs in the world

`world.GetAllSwarms()` <- Gets all Swarms in the world

`world.Sync()` <- Syncs swarms to world

`Swarm.GetAllUAVs()` <- Gets all UAVs in the world

```
import math


a=10  #drones in swarm A
b=20  #drones in swarm B
c=10  #drones in swarm C
d_ab=np.random.randint(1,50,size=(10,10)) # dist bw swarm A nad B
d_bc=np.random.randint(1,50,size=(10,10)) #dist bw swarm B and C

p=20
g=30
f=28
noise=-174

# PATH LOSS
pl_ab = np.zeros((10,10))
pl_bc =  np.zeros((10,10))

def pl(d,f):
    a=np.log10(d)+np.log10(f)
    b=20*a
    return b

for i in range (0,10):
  for j in range (0,10):
    pl_ab[i,j]=pl(d_ab[i,j],f)
    pl_bc[i,j]=pl(d_bc[i][j],f)
```

n = [50, 100]

```
ALGORITHM 1: Render max min output.

Init N ←min(size(c))

c’ ←c

While N>0 do

    c’[i][j] ←max(c’);

    c’ ← remove row i and column j of c’ ;

    Sij ←1;

    N reduce 1;

σ ← S × c × tk /m;
```

```
# Algo 1 Max min throughput
# N = 50.0

c_prime = []
def MxMnThroughput( matrix, N ) :
    while( N > 0 ) :

      if( matrix.shape == (1,1) ) : break

      c_prime_max = matrix.max()
      idx = np.where( matrix == c_prime_max )[0][0]
      idy = np.where( matrix == c_prime_max )[1][0]

      matrix = np.delete( matrix, idx , 0 )
      matrix = np.delete( matrix, idy , 1 )

      N -= 1

    return matrix
```

```
d_ab
```

```
array([[47, 30, 26, 20,  9, 20,  3, 48, 16, 28],
       [30, 45,  7,  2, 27,  4, 19, 20, 12, 21],
       [ 4,  4, 39, 11, 40, 44,  5, 48,  9, 40],
       [26, 20, 18, 21, 23, 49, 47, 47, 11, 44],
       [40,  9,  4,  0, 10, 25, 13, 43, 19,  2],
       [48, 44, 18, 15, 14, 19, 12, 16, 46, 14],
       [ 6, 17, 22, 45, 42,  5, 16, 48, 49, 23],
       [38, 48, 15,  5, 48, 26, 45,  2,  2, 44],
       [ 6, 41, 11, 40, 13, 26, 21, 35,  2, 26],
       [34, 22, 41, 49, 22, 35, 33, 46, 41, 46]])
```

```
c_t = d_ab
```

```
# [ MxMnThroughput(pl_ab , i) for i in range( 20, 51, 10 ) ]

throughput1 = MxMnThroughput( pl_ab, 12 )[0][0]
```

```
m_list = [ m for m in range( 45, 75 ) ]
```

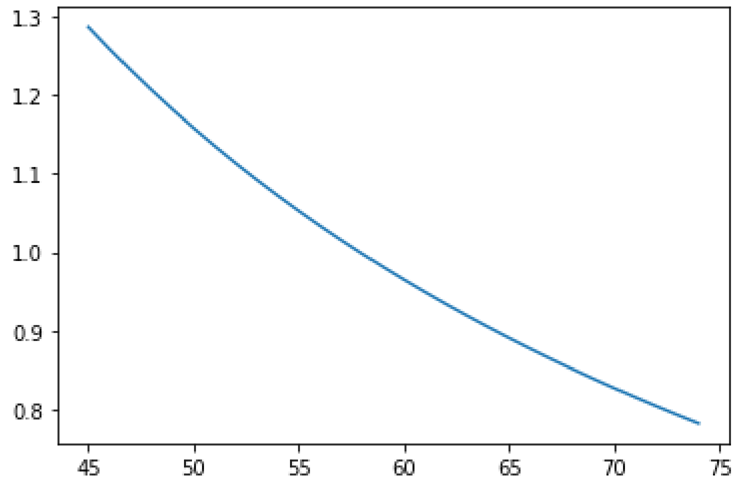
```
throughput_list = [ throughput1 / m for m in m_list ]
```

```
throughput_list
```

```
[1.286362694526417,
 1.2583982881236688,
 1.2316238564614632,
 1.205965026118516,
 1.1813534949732403,
 1.1577264250737753,
 1.1350259069350739,
 1.113198485647861,
 1.0921947406356372,
 1.0719689121053475,
 1.0524785682488866,
 1.0336843081015852,
 1.0155494956787503,
 0.9980400216153236,
 0.9811240890455724,
 0.9647720208948128,
 0.9489560861260454,
 0.9336503428014318,
 0.918830496090298,
 0.904473769588887,
 0.8905587885182887,
 0.8770654735407389,
 0.863974944084907,
 0.8512694302013054,
 0.838932192082446,
 0.8269474464812682,
 0.8153002993477292,
 0.8039766840790107,
 0.7929633048450516,
 0.7822475845093078]
```

```
sus.lineplot( m_list, throughput_list )
```

```
/usr/local/lib/python3.8/dist-packages/seaborn/_decorators.py:36: FutureWarning: Pas
  warnings.warn(
<matplotlib.axes._subplots.AxesSubplot at 0x7fc5e6a55100>
```


1.3
1.2
1.1
1.0
0.9
0.8
45
50
55
60
65
70
75