

```
In [10]: 1 import sympy
2 import random
3 import matplotlib.pyplot as pp
4 import numpy as np
5
6 %matplotlib inline
```

Parametros

```
In [11]: 1 HOSPITALES = 2
2 HOSPITAL_CAMA_A = 10
3 HOSPITAL_CAMA_B = 10
4 HOSPITAL_VENT_A = 4
5 HOSPITAL_VENT_B = 4
6 INFECTADOS = 20
7 TASA_CRECIMIENTO = 7
8 PERSONAL_MEDICO = 150
9 DIAS_INTERNADO = 10
10 DIAS_SIMULACION = 20
11
12
13 persona_recuperadas={}
14 persona_fallecidas={}
```

Metodos para somular la hospitalizacion

```

In [12]: 1 class Hospital(object):
2         def __init__(self, env, num_cama, name):
3             self.env = env
4             self.num_cama = num_cama
5             self.camas = simpy.Resource(env, num_cama)
6             self.name = name
7
8         def ingresar_paciente(self, paciente):
9             yield self.env.timeout(random.randint(DIAS_INTERNADO-5, DIAS_INTERNADO+5))
10            print("El paciente se termino de atender: ", paciente, " tiempo de salida: ", self.env.now, " ho
11
12 def llegada_paciente(env, hospital, paciente):
13     arrive = env.now
14     estado = random.randint(1,100)
15     if (estado < 60):
16         with hospital.camas.request() as cama:
17             dias_esperando = random.randint(1,5)
18             requerimiento = yield cama | env.timeout(dias_esperando)
19             wait = env.now - arrive
20             if cama in requerimiento:
21                 print("Al paciente: ", paciente, " se le asigna una CAMA ", " hospital " , hospital.name)
22                 yield env.process(hospital.ingresar_paciente(paciente))
23                 estado = random.randint(1,100)
24                 if (estado < 8) :
25                     persona_fallecidas[env.now] = persona_fallecidas[env.now] + 1 if env.now in persona_
26                 else:
27                     persona_recuperadas[env.now] = persona_recuperadas[env.now] + 1 if env.now in person
28             else:
29                 print("El paciente " , paciente, " en el hospital ", hospital.name, " espero ", wait , '
30                 persona_fallecidas[env.now] = persona_fallecidas[env.now] + 1 if env.now in persona_fall
31         else :
32             print("El paciente no tiene COVID : ", paciente, " hospital ", hospital.name )
33
34 def ejecutar(env, tasa_crecimiento, infectados):
35     hospitalA = Hospital(env, HOSPITAL_CAMA_A, "A")
36     hospitalB = Hospital(env, HOSPITAL_CAMA_B, "B")
37     for i in range(infectados):
38         asignar_hospital(env, hospitalA, hospitalB, i)
39     paciente = infectados
40     while True:
41         yield env.timeout(1)
42         for i in range(tasa_crecimiento):

```

```

43         paciente += 1
44         asignar_hospital(env, hospitalA, hospitalB, paciente)
45
46 def asignar_hospital(env, hospitalA, hospitalB, paciente):
47     hosp_esc = random.randint(1,2)
48     if (hosp_esc == 1):
49         print("Llega el paciente ", paciente, " al hospital A en el tiempo", env.now)
50         env.process(llegada_paciente(env, hospitalA, paciente))
51     else:
52         print("Llega el paciente ", paciente, " al hospital B en el tiempo ", env.now)
53         env.process(llegada_paciente(env, hospitalB, paciente))

```

Iniciamos Simulacion

```

In [13]: 1 print("Simulacion Hospitalizacion")
          2 env=simpy.Environment()
          3 env.process(ejecutar(env,TASA_CRECIMIENTO, INFECTADOS))
          4 env.run(until=DIAS_SIMULACION)

```

Llega el paciente 131 al hospital B en el tiempo 16
 Llega el paciente 132 al hospital B en el tiempo 16
 El paciente no tiene COVID : 130 hospital A
 El paciente no tiene COVID : 131 hospital B
 El paciente no tiene COVID : 132 hospital B
 El paciente se termino de atender: 38 tiempo de salida: 17 hospital A
 El paciente se termino de atender: 66 tiempo de salida: 17 hospital B
 El paciente se termino de atender: 80 tiempo de salida: 17 hospital B
 El paciente se termino de atender: 89 tiempo de salida: 17 hospital A

Llega el paciente 133 al hospital A en el tiempo 17
 Llega el paciente 134 al hospital A en el tiempo 17
 Llega el paciente 135 al hospital A en el tiempo 17
 Llega el paciente 136 al hospital B en el tiempo 17
 Llega el paciente 137 al hospital A en el tiempo 17
 Llega el paciente 138 al hospital B en el tiempo 17
 Llega el paciente 139 al hospital A en el tiempo 17
 El paciente no tiene COVID : 133 hospital A
 El paciente no tiene COVID : 135 hospital A
 El paciente no tiene COVID : 136 hospital B
 El paciente no tiene COVID : 137 hospital A

Resultados y graficos

```
In [14]: 1 print("Resultados pacientes :")
          2 print("Recuperados: ")
          3 print(persona_recuperadas)
          4 print("Fallecidos: ")
          5 print(persona_fallecidas)
```

Resultados pacientes :

Recuperados:

{7: 3, 8: 3, 11: 2, 12: 4, 13: 3, 14: 2, 15: 2, 17: 4, 18: 3, 19: 3}

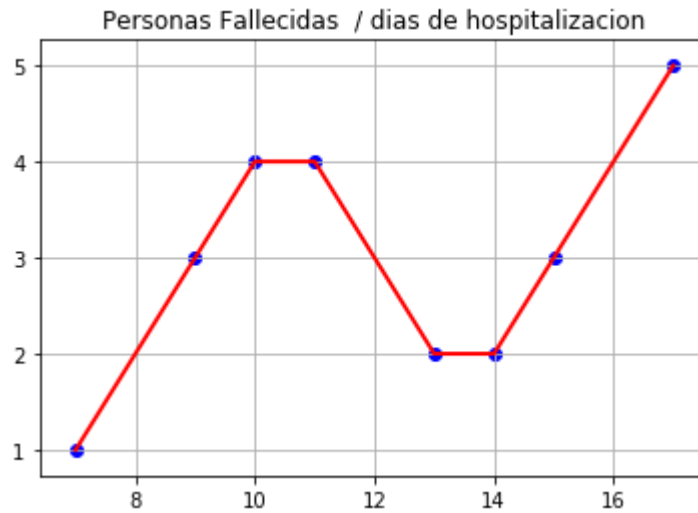
Fallecidos:

{7: 1, 9: 3, 10: 4, 11: 4, 13: 2, 14: 2, 15: 3, 17: 5}

```
In [16]: 1 datos=sorted(persona_recuperadas.items())
          2 x, y =zip(*datos)
          3 pp.plot(x,y,linewidth=2,color='red')
          4 pp.scatter(x,y,color='blue')
          5 pp.title("Dias / Personas Recuperadas")
          6 pp.grid(True)
          7 pp.show()
```



```
In [17]: 1 if (persona_fallecidas):  
2     datos=sorted(persona_fallecidas.items())  
3     x, y =zip(*datos)  
4     pp.plot(x,y,linewidth=2,color='red')  
5     pp.scatter(x,y,color='blue')  
6     pp.title("Personas Fallecidas / dias de hospitalizacion")  
7     pp.grid(True)  
8     pp.show()
```



```
In [ ]: 1
```