

Universidad Politécnica Salesiana

Nombre: Angel Jadan

Fecha: 19/07/2021

Ejercicio:

Utilizando las tarea de la predicción de llegadas de vacunas y el recinto de vacunación, realizar un sistema que permita simular y correlacionar el procesos de llegada/compras de vacuna con el procesos de vacunación, en donde si no se tiene un stock/número de vacunas las personas tendran que esperar/reasignar a otro día en donde exista vacunas dentro del establecimiento y realizar el proceso de vacunación

```
pip install simpy
```

```
Collecting simpy
  Downloading simpy-4.0.1-py2.py3-none-any.whl (29 kB)
Installing collected packages: simpy
Successfully installed simpy-4.0.1
```

```
import simpy
```

```
def llegada_vacuna(env, bodega):
    for i in range(1000):
        yield env.timeout(2)
        yield bodega.put(f'vacuna{i}')
        print(f'Llego la vacuna {i} en el tiempo {env.now}')
```

```
def paciente(nombre,env,bodega):
    while True:
        yield env.timeout(1)
        print(f'{nombre} se coloca la vacuna en el tiempo {env.now}')
env = simpy.Environment()
bodega = simpy.Store(env, capacity=1)
llegada = env.process(llegada_vacuna(env, bodega))
paciente = env.process(paciente("1",env, bodega))
env.run(until=10)
```

```
1 se coloca la vacuna en el tiempo 1
1 se coloca la vacuna en el tiempo 2
Llego la vacuna 0 en el tiempo 2
1 se coloca la vacuna en el tiempo 3
1 se coloca la vacuna en el tiempo 4
1 se coloca la vacuna en el tiempo 5
1 se coloca la vacuna en el tiempo 6
1 se coloca la vacuna en el tiempo 7
1 se coloca la vacuna en el tiempo 8
1 se coloca la vacuna en el tiempo 9
```

```
def vacuna_terminada(env, problemas):
    for problema in [simpy.PriorityItem('P2','#0000'),simpy.PriorityItem('P0','#0001'),simpy.PriorityItem('P3','#0002')]:
        yield env.timeout(1)
        print(f'En el tiempo {env.now} se terminarán las vacunas {problema}')
        yield problemas.put(problema)
```

```
def cambio_punto(env,problemas):
    while True:
        problema = yield problemas.get()
        yield env.timeout(3)
        print(f'Se ha cambiado al paciente {problema} en el tiempo {env.now}')
```

```
env = simpy.Environment()
problemas = simpy.PriorityStore(env)
env.process(vacuna_terminada(env.problemas))
```

```
env.process(cambio_punto(env, problemas))  
env.run(until=20)
```

```
En el tiempo 1 se terminaron las vacunas PriorityItem(priority='P2', item='#0000')  
En el tiempo 2 se terminaron las vacunas PriorityItem(priority='P0', item='#0001')  
En el tiempo 3 se terminaron las vacunas PriorityItem(priority='P3', item='#0002')  
Se ha cambiado al paciente PriorityItem(priority='P2', item='#0000') en el tiempo 4  
En el tiempo 4 se terminaron las vacunas PriorityItem(priority='P1', item='#0003')  
Se ha cambiado al paciente PriorityItem(priority='P0', item='#0001') en el tiempo 7  
Se ha cambiado al paciente PriorityItem(priority='P1', item='#0003') en el tiempo 10  
Se ha cambiado al paciente PriorityItem(priority='P3', item='#0002') en el tiempo 13
```

✓ 0 s completado a las 16:07

