

# Practica No 13: Peliculas

Nombre:

Fernando Sanchez

Enunciado:

Al modelo de simulación de la venta de boletos de un teatro /cine agregar la simulación grafica 2D/3D utilizando cualquier libreria grafica, por ejemplo pygame/tkinter, etc. generar un informe en formato PDF que me permita evidenciar la simulación y generar una grafica/reporte de los resultados.

```
In [9]: 1 # Importar las librerias para el analisis
2 import collections
3 import random
4 import simpy
5 from tkinter import *
6 from tkinter import messagebox
7 from tkinter import ttk
8 import tkinter as tk
```

```
In [3]: 1 def ventaBoletos(env, num_boletos, pelicula, teatro):
2     with teatro.contador.request() as turno:
3         resultado = yield turno | teatro.sold_out[pelicula]
4         if turno not in resultado:
5             teatro.num_renegados[pelicula] += 1
6             return
7         if teatro.num_boletos[pelicula] < num_boletos:
8             yield env.timeout(0.5)
9             return
10        teatro.num_boletos[pelicula] -= num_boletos
11        if teatro.num_boletos[pelicula] < 2:
12            teatro.sold_out[pelicula].succeed()
13            teatro.tiempo_agotado[pelicula] = env.now
14            teatro.num_boletos[pelicula] = 0
15        yield env.timeout(1)
```

```
In [4]: 1 def llegadaClientes(env, teatro):
2     while True:
3         yield env.timeout(random.expovariate(1/0.5))
4         pelicula = random.choices(teatro.peliculas, teatro.prob
5         num_boletos = random.randint(1, 10)
6         if teatro.num_boletos[pelicula[0]]:
7             env.process(ventaBoletos(env, num_boletos, pelicula
```

```

In [5]: 1 def process_run(velocidad,cantidadBoleto,tmSimulacion):
2         VELOCIDAD_LLEGADA = velocidad
3         NUM_BOLETO = cantidadBoleto
4         TIEMPO_SIMULACION = tmSimulacion
5
6         resultado = ""
7
8         Teatro = collections.namedtuple('Teatro', 'contador, peliculas')
9
10        print('Teatro Carlos Crespi - UPS')
11
12        resultado = "Teatro Carlos Crespi - UPS"
13        env = simpy.Environment()
14
15        contador = simpy.Resource(env,capacity=1)
16        peliculas = ['Conjuro 3', 'Rápidos y Furiosos 10', 'Pulp Fiction']
17
18        resultado = resultado + "\n"
19
20        probabilidad=[0.1, 0.3, 0.5, 0.7, 0.9]
21        num_boletos = {pelicula: NUM_BOLETO for pelicula in peliculas}
22        sold_out = {pelicula: env.event() for pelicula in peliculas}
23        tiempo_agotado = {pelicula: None for pelicula in peliculas}
24        num_renegados = {pelicula: 0 for pelicula in peliculas}
25
26        teatro = Teatro(contador, peliculas, probabilidad, num_boletos)
27        env.process(llegadaClientes(env, teatro))
28        env.run(until=TIEMPO_SIMULACION)
29
30        # Analisis y resultados
31        for pelicula in peliculas:
32
33            if teatro.sold_out[pelicula]:
34
35                print('Pelicula: %s se agoto en el tiempo %.1f desp' % (pelicula, tiempo_agotado[pelicula]))
36                sms1 = 'Pelicula: %s se agoto en el tiempo %.1f desp' % (pelicula, tiempo_agotado[pelicula])
37
38                print('Numero de personas que salieron de la fila/r' % (pelicula, num_renegados[pelicula]))
39                sms2 = 'Numero de personas que salieron de la fila/r' % (pelicula, num_renegados[pelicula])
40
41                resultado = resultado + "\n" + sms1 + "\n" + sms2
42        return resultado

```

```
In [7]: 1 resp = process_run(velocidad=52,cantidadBoleto=18,tmSimulacion=
2 print("Resultado=> ",resp)
```

Teatro Carlos Crespi – UPS

Pelicula: Conjuro 3 se agoto en el tiempo 108.9 despues de salir a la venta

Numero de personas que salieron de la fila/renegados 0

Pelicula: Rapidos y Furiosos 10 se agoto en el tiempo 23.4 despues de salir a la venta

Numero de personas que salieron de la fila/renegados 0

Pelicula: Pulp Fictions se agoto en el tiempo 16.4 despues de salir a la venta

Numero de personas que salieron de la fila/renegados 3

Pelicula: La liga de la Justicia se agoto en el tiempo 13.4 despues de salir a la venta

Numero de personas que salieron de la fila/renegados 3

Pelicula: Godzilla vs Kong se agoto en el tiempo 14.9 despues de salir a la venta

Numero de personas que salieron de la fila/renegados 6

Resultado=> Teatro Carlos Crespi – UPS

Pelicula: Conjuro 3 se agoto en el tiempo 108.9 despues de salir a la venta

Numero de personas que salieron de la fila/renegados 0

Pelicula: Rapidos y Furiosos 10 se agoto en el tiempo 23.4 despues de salir a la venta

Numero de personas que salieron de la fila/renegados 0

Pelicula: Pulp Fictions se agoto en el tiempo 16.4 despues de salir a la venta

Numero de personas que salieron de la fila/renegados 3

Pelicula: La liga de la Justicia se agoto en el tiempo 13.4 despues de salir a la venta

Numero de personas que salieron de la fila/renegados 3

Pelicula: Godzilla vs Kong se agoto en el tiempo 14.9 despues de salir a la venta

Numero de personas que salieron de la fila/renegados 6

```

In [15]: 1 grafica = tk.Tk()
2 grafica.geometry("350x200")
3 grafica.title("Simulación de películas")
4 #grafica.config(bg = 'powder blue')
5 frame =Frame (bg = 'black')
6 grafica.resizable(0,0)
7
8 txtVelocidad=StringVar()
9 velocidad = Label(grafica,text="Velocidad", fg='black', width=10)
10 lblVelocidad = ttk.Entry(grafica, textvariable=txtVelocidad).place(x=10, y=10)
11
12 txtBoleto=StringVar()
13 cantidadBoleto = Label( grafica,text="Cantida boletos", fg='black', width=10)
14 lblBoleto = ttk.Entry(grafica, textvariable=txtBoleto).place(x=10, y=30)
15
16 txtSimulacion=StringVar()
17 simulacion = Label(grafica,text="Tiempo de simulacion", fg='black', width=10)
18 lblSimulacion = ttk.Entry(grafica,textvariable=txtSimulacion).place(x=10, y=50)
19
20 def guardar():
21     velocida = txtVelocidad.get()
22     boletos=txtBoleto.get()
23     simulacions=txtSimulacion.get()
24
25     resultado = process_run(velocidas=int(velocida), cantidadBoletos=int(boletos), tiempo=int(simulacions))
26     print("resp ",resultado)
27     messagebox.showinfo(message=resultado,title="Datos")
28
29
30 Button(grafica, text="Ejecutar", bg='gray', fg='black', command=guardar).place(x=10, y=70)
31
32 grafica.mainloop()
33

```

```

In [ ]: 1

```