UNIVERSIDAD POLITECNICA SALESIANA

SEDE CUENCA

CARRERA: *INGENIERIA DE SISTEMAS*

Nombre: Bryam Gabriel Mora Lituma

Materia: Simulacion

Fecha:04/07/2021

Diseñe y desarrolle un modelo y/o script que permita simular el siguiente caso real: •En base a los datos del siguiente link https://educacion.gob.ec/wp-content/uploads/downloads/2012/08/AZUAY11.pdf (https://educacion.gob.ec/wp-content/uploads/downloads/2012/08/AZUAY11.pdf), genere una simulación del ingresos de los estudiantes, para ello debemos escoger un establecimiento y en base a los docentes y estudiantes modelar el reingreso de los estudiantes en base a los siguientes datos.

◦ Solo se va a tener en cuenta uno de los planteles educativos(Escuela, colegio, universidad dentro del Azuay).

◦ Se tiene un promedio que el 90% de los docentes han sido vacunados y pueden realizar el proceso de ingreso en cada uno de los cursos.

◦ Dentro del procesos se tiene que alrededor del 5% - 10% de los estudiantes no podrán asistir debido a no presentar la vacuna/enfermedades adyacentes.

◦ Los estudiantes solo pertenecen a una sola entidad educativa al igual que los docentes.

◦ Se va a tener un periodo de prueba de un mes, posterior a ello se realiza al azar al 10% de estudiantes una prueba PCR para validar que no estén contagiados.

◦ De la ultima el 2% de los estudiantes dan positivo por lo que se cierra el curso completo.

◦ Los estudiantes asisten cada semana y estos están en un horario de 6 horas ya sea diurno o nocturno.

◦ Tienen un receso 30 minutos dentro del establecimiento en donde se concentran todos los estudiantes y exite un foco de contagio del 2%.

• El proceso de simulación desarrollado deberá considerar los siguientes aspectos:

◦ Generar un cuaderno de Python para el desarrollo y parametrización de graficas, reportes, y animación (Simpy).

◦ Generar una animación en 2D/3D del modelo propuesto.

◦ Obtener los siguientes análisis:

◦ Cuantos contagiados tengo al final del mes.

◦ Cuantos cursos debo cerrar.

◦ Cuantos estudiantes y docentes ingresan y salen al final del mes.

Unidad Educativa: Sudamericano

Estudiantes.

Número total de estudiantes de genero femenino: 212

Número total de estudiantes de genero masculino: 350

Número total de estudiantes del establecimiento: 562

Docentes.

Número de Docentes genero femenino: 24

Número de Docentes genero masculino: 16

Número total de profesores: 40

Aulas: 30

https://www.infoescuelas.com/ecuador/azuay/unidad-educativa-particular-sudamericano-en-cuenca/ (https://www.infoescuelas.com/ecuador/azuay/unidad-educativa-particular-sudamericano-en-cuenca/)

Importamos las librerias

In [100]:

```
import random
import simpy
import random
import collections
import matplotlib.pyplot as pp
```

Iniciamamos nuestras variables

In [101]:

```
PROFESORES = 40
P_PROF_VACUNADOS = int(PROFESORES*0.9)
VACUNADOS_PROF = PROFESORES - P_PROF_VACUNADOS
print("Profesores: ",PROFESORES)
print("Profesores vacunados: ",P_PROF_VACUNADOS)
print("Profesores no vacunados:",VACUNADOS_PROF)
ESTUDIANTES = 562
print("Estudiantes: ",ESTUDIANTES)
P_ESTUDIANTES_NO = random.randint(5,10)
print("Porcentaje de estudiantes que no asisten: ",P_ESTUDIANTES_NO,"%")
ESTUDIANTES_NO_ASISTEN = int(ESTUDIANTES * P_ESTUDIANTES_NO /100)
print("Numero de estudiantes que no asisten: ",ESTUDIANTES_NO_ASISTEN)
ESTUDIANTES_ASISTEN = ESTUDIANTES-ESTUDIANTES_NO_ASISTEN
print("Numero de estudiantes que asisten: ",ESTUDIANTES_ASISTEN)

#Total de estudiantes y profesores vacunados que asisten
TOTAL = P_PROF_VACUNADOS + ESTUDIANTES_ASISTEN

TOTAL_CONTAGIADOS = 0

print("Total de profesores y estudiantes tanto vacunados y que asisten: ",TOTAL)

cursos = ['curso 1','curso 2','curso 3','curso 4','curso 5','curso 6']
aula1 = {}
aula2 = {}
aula3 = {}
aula4 = {}
aula5 = {}
aula6 = {}

#print("Cursos: ",cursos)

#Para el numero de contagiados al final del
contagiados = []
```

```
Profesores:  40
Profesores vacunados:  36
Profesores no vacunados: 4
Estudiantes:  562
Porcentaje de estudiantes que no asisten:  5 %
Numero de estudiantes que no asisten:  28
```

```
Numero de estudiantes que asisten:  534
Total de profesores y estudiantes tanto vacunados y que asisten:  570
```

Metodos para asignar el aula y la clase

In [102]:

```
def uso_aula(env,estudiante,aula,curso):
    print(aula)
    print("Hora Ingreso",env.now)
    if aula=="Curso 1":
        aula1[estudiante]=estudiante
        yield env.timeout(0.01)
        return
    if aula=="Curso 2":
        aula2[estudiante]=estudiante
        yield env.timeout(0.01)
        return
    if aula=="Curso 3":
        aula3[estudiante]=estudiante
        yield env.timeout(0.01)
        return
    if aula=="Curso 4":
        aula4[estudiante]=estudiante
        yield env.timeout(0.01)
        return
    if aula=="Curso 5":
        aula5[estudiante]=estudiante
        yield env.timeout(0.01)
        return
    if aula=="Curso 6":
        aula6[estudiante]=estudiante
        yield env.timeout(0.01)
        return
    yield env.timeout(0.05)
    return

def clases(env, curso):
    print("Clases")
    c = 0
    while True:
        yield env.timeout(random.expovariate(1/0.005))
        print("time ",env.now)
        aula = random.choice(curso.cursos)
        print("El estudiante pertenece al curso. ",aula)
        print("Hora: ",env.now)
        estudiante = curso.estudiantes[c]
        print("Estudiante. ",estudiante)
        tot = len(aula1)+len(aula2)+len(aula3)+len(aula4)+len(aula5)+len(aula6)
```

```
        if tot < curso.n_estudiantes:
            yield env.process(uso_aula(env, estudiante,aula,curso))
        else:
            print("Ya no hay mas estudiantes")
            print("Tiempo ",env.now)
            break
        c+=1
```

In [103]:

```
Curso = collections.namedtuple('Curso','contador,cursos,capacidad,usado,t_curso_lleno,puesto_usado,estuc
print("Institucion educativa SUDAMERICANO")

env = simpy.Environment()

contador = simpy.Resource(env, capacity=6)
cursos = ["Curso 1","Curso 2","Curso 3","Curso 4","Curso 5","Curso 6"]
capacidad = {curso: int(((ESTUDIANTES_ASISTEN)/len(cursos))+3) for curso in cursos}
usado = {curso:env.event() for curso in cursos}
#Tiempo en el que el curso se llena
t_curso_lleno = {curso: None for curso in cursos}

#Numero de puestos usados
puesto_usado = {curso: 0 for curso in cursos}

estudiantes =[0] * ESTUDIANTES_ASISTEN
for j in range(ESTUDIANTES_ASISTEN):
    estudiantes[j] = "Estudiante "+str(j)

n_estudiantes = ESTUDIANTES_ASISTEN

print("cursos: ",cursos[1])

curs = Curso(contador, cursos, capacidad, usado,t_curso_lleno,puesto_usado,estudiantes,n_estudiantes)
env.process(clases(env,curs))
env.run(until=35)
```

```
Hora:  5.776702033787296
Estudiante.  Estudiante 390

Curso 6
Hora Ingreso 5.776702033787296
time  5.787169008918901
El estudiante pertenece al curso.  Curso 1
Hora:  5.787169008918901
Estudiante.  Estudiante 391
Curso 1
Hora Ingreso 5.787169008918901
time  5.808187358459898
El estudiante pertenece al curso.  Curso 6
Hora:  5.808187358459898
Estudiante.  Estudiante 392
Curso 6
```

```
Hora Ingreso 5.808187358459898
time  5.819424010560729
El estudiante pertenece al curso.  Curso 2
Hora:  5.819424010560729
```

Estudiantes asignados por aula

In [104]:

```
tot = len(aula1)+len(aula2)+len(aula3)+len(aula4)+len(aula5)+len(aula6)

aulas_x = ["aula 1","aula 2","aula 3","aula 4","aula 5","aula 6"]
aulas_y = [len(aula1), len(aula2), len(aula3), len(aula4), len(aula5), len(aula6)]
print("Total estudiantes que deben asistir en el periodo de prueba. ",n_estudiantes)

print("Estudiantes por aula")
print("Aula 1 ",len(aula1))
print("Aula 2 ",len(aula2))
print("Aula 3 ",len(aula3))
print("Aula 4 ",len(aula4))
print("Aula 5 ",len(aula5))
print("Aula 6 ",len(aula6))
print("_____________________________")
print("Total de \nestudiantes que asistierón ",tot)
```

```
Total estudiantes que deben asistir en el periodo de prueba.  534
Estudiantes por aula
Aula 1  79
Aula 2  92
Aula 3  82
Aula 4  96
Aula 5  92
Aula 6  93
_____________________________
Total de
estudiantes que asistierón  534
```

Estudiantes contagiados y aula cerrada

In [105]:

```
por_pcr = int(tot*0.10)
print("Numero de estudiantes ha hacer la prueba: ",por_pcr," de ",tot)
por_pcr = int(len(aula6)*0.02)
print("EL numero de aula 6 se cierra")
print("Numero de estudiantes que dan positivo de la ultima aula: ",por_pcr," de ",len(aula6))
TOTAL_CONTAGIADOS=TOTAL_CONTAGIADOS+por_pcr
print("TOTAL CONTAGIADOS: ",TOTAL_CONTAGIADOS)
```

```
Numero de estudiantes ha hacer la prueba:  53  de  534
Aula 6 se cierra
Numero de estudiantes que dan positivo de la ultima aula:  1  de  93
TOTAL CONTAGIADOS:  1
```

Grafica de los estudiantes con relacion al numero de aulas

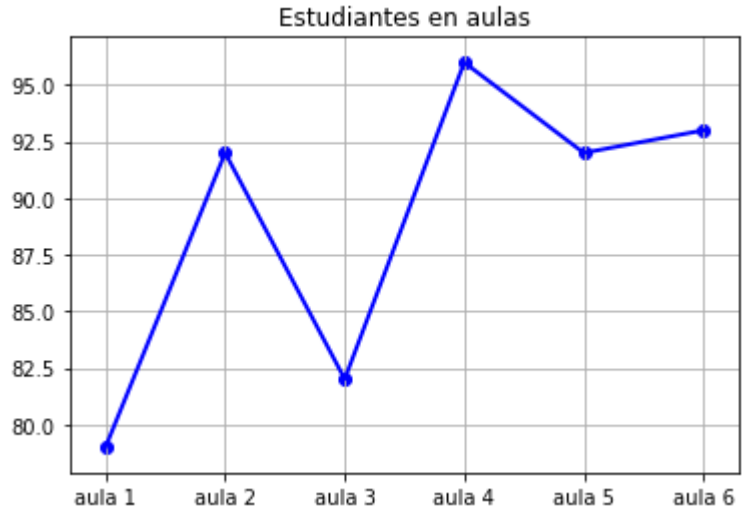
In [106]:

```
x = aulas_x
y = aulas_y

print(x)
print(y)

pp.plot(x,y, linewidth=2,color='blue')
pp.scatter(x,y,color='blue')
pp.title("Estudiantes en aulas")
pp.grid(True)
pp.show()
```

```
['aula 1', 'aula 2', 'aula 3', 'aula 4', 'aula 5', 'aula 6']
[79, 92, 82, 96, 92, 93]
```



Metodos para simular el porceos de regreso a clases

In [107]:

```
def inicio_clases(env, resource, espera, prio):
    yield env.timeout(espera)
    with resource.request(priority=prio) as req:
        yield req
        print("Inicio clases: ",env.now)
        yield env.timeout(3.5)
        return
    print("Fin primera jornada: ",env.now)
    return

def receso(env,n_est,resource,espera,prio):
    yield env.timeout(espera)
    with resource.request(priority=prio) as req:
        print("Inicio de receso: ",env.now)
        yield req
        print("est",n_est)
        contg = int(n_est * 0.02)
        contg = contg
        print("Numero de contagiados 2% en receso. ",contg)
        yield env.timeout(0.5)#0.5 representa media hora (30 min)
        return
    print("Fin receso: ",env.now)
    return

def fin_clases(env, resource, espera, prio):
    yield env.timeout(espera)
    with resource.request(priority=prio) as req:
        yield req
        print("Inicio segunda jornada: ",env.now)
        yield env.timeout(2)
        return
    print("Fin segunda jornada: ",env.now)
    return
```

Resultado del proceso

In [108]:
```
env = simpy.Environment()
res = simpy.PriorityResource(env, capacity=1)

t_est = len(aula1)+len(aula2)+len(aula3)+len(aula4)+len(aula5)
#print(t_est)
jornada1 = env.process(inicio_clases(env,res, espera=0,prio=0))
receso = env.process(receso(env,t_est,res, espera=3.5,prio=1))
jornada2 = env.process(fin_clases(env,res, espera=2,prio=2))

env.run()
```

```
441
Inicio clases:  0
Inicio de receso:  3.5
est 441
Numero de contagiados 2% en receso.  8
Inicio segunda jornada:  4.0
```

Estudiantes contagiados durante el regreso a clases

In [109]:
```
contg = int(t_est*0.02)
t=contg+TOTAL_CONTAGIADOS
print("Número de estudiantes de última aula cerrada: ",len(aula6))
print("Numero de contagiados con prueba pcr: ",TOTAL_CONTAGIADOS)
print("Numero de contagiados en el receso de 4 aulas: ",contg)
print("Numero de contagiados en total: ",t)
```

```
Número de estudiantes de última aula cerrada:  93
Numero de contagiados con prueba pcr:  1
Numero de contagiados en el receso de 4 aulas:  8
Numero de contagiados en total:  9
```

In [ ]:
```

```