

In [50]:

```
1 import random
2 import simpy
3 import random
4 import collections
```

In [68]:

```
1 print("Escuela básica o bachillerato LA 'ASUNCIÓN' ")
2 print("    Númeo de estudiantes")
3 print("*****Hombres: 1004")
4 print("*****Mujeres: 1691")
5 print("    Total: 2695")
6
7 PROFESORES = 145
8 P_PROF_VACUNADOS = int(PROFESORES*0.9)
9 VACUNADOS_PROF = PROFESORES - P_PROF_VACUNADOS
10 print("Profesores: ",PROFESORES)
11 print("Profesores vacunados: ",P_PROF_VACUNADOS)
12 print("Profesores no vacunados:",VACUNADOS_PROF)
13 ESTUDIANTES = 2695
14 print("Estudiantes: ",ESTUDIANTES)
15 P_ESTUDIANTES_NO = random.randint(5,10)
16 print("% de estudiantes que no asisten: ",P_ESTUDIANTES_NO,"%")
17 ESTUDIANTES_NO_ASISTEN = int(ESTUDIANTES * P_ESTUDIANTES_NO /100)
18 print("Nº Estudiantes que no asisten: ",ESTUDIANTES_NO_ASISTEN)
19 ESTUDIANTES_ASISTEN = ESTUDIANTES-ESTUDIANTES_NO_ASISTEN
20 print("Estudiantes que asisten: ",ESTUDIANTES_ASISTEN)
21
22 TOTAL = P_PROF_VACUNADOS + ESTUDIANTES_ASISTEN
23 print("total: ",TOTAL)
24
25 cursos = ['curso 1','curso 2','curso 3','curso 4','curso 5','curso 6']
26 print("Cursos: ",cursos)
27
28 #Para el numero de contagiados al final del
29 contagiados = []
```

Escuela básica o bachillerato LA 'ASUNCIÓN'

Númeo de estudiantes

\*\*\*\*\*Hombres: 1004

\*\*\*\*\*Mujeres: 1691

Total: 2695

Profesores: 145

Profesores vacunados: 130

Profesores no vacunados: 15

Estudiantes: 2695

% de estudiantes que no asisten: 7 %

Nº Estudiantes que no asisten: 188

Estudiantes que asisten: 2507

total: 2637

Cursos: ['curso 1', 'curso 2', 'curso 3', 'curso 4', 'curso 5', 'curso 6']

In [52]:

```
1 def p_prueba(env, curso):
2     print("inicia periodo de prueba")
```

In [90]:

```
1 def uso_aula(env, estudiantes, aula, curso):
2     with curso.contador.request() as uso:
3         res = yield uso | curso.prueba[aula]
4         print(res)
5         return
6
7 def clases(env, curso):
8     print("Clases")
9     while True:
10         yield env.timeout(random.expovariate(1/0.5))
11
12     for i in range(len(curso.cursos)):
13         aula = curso.cursos[i]
14         print("aula", aula)
15         estudiantes = 10
16         env.process(uso_aula(env, estudiantes, aula, curso))
```

In [91]:

```
1 Curso = collections.namedtuple('Curso', 'contador,cursos,capacidad,prueba,t_curso_lleno,
2 print("Institucion educativa")
3
4 env = simpy.Environment()
5
6 contador = simpy.Resource(env, capacity=1)
7 cursos = ["Curso 1","Curso 2","Curso 3","Curso 4","Curso 5","Curso 6"]
8 capacidad = {curso: int(((TOTAL)/len(cursos))+3) for curso in cursos}
9 prueba = {curso:env.event() for curso in cursos}
10 t_curso_lleno = {curso: None for curso in cursos}
11 generados = {curso: 0 for curso in cursos}
12
13 print("cur: ",cursos[1])
14 print("capacidad: ",capacidad)
15 print("prueba: ",prueba)
16 print("t_curso_llenado: ", t_curso_lleno)
17 print("generados: ",generados)
18
19 curs = Curso(contador, cursos, capacidad, prueba,t_curso_lleno,generados)
20 env.process(clases(env,curs))
21 env.run(until=20)
```

<ConditionValue {<Request() object at 0x175fd5c8ac0>: None}>

aula Curso 1

aula Curso 2

aula Curso 3

aula Curso 4

aula Curso 5

aula Curso 6

<ConditionValue {<Request() object at 0x175fa2c7e80>: None}>

<ConditionValue {<Request() object at 0x175fd5c5730>: None}>

<ConditionValue {<Request() object at 0x175fd503550>: None}>

<ConditionValue {<Request() object at 0x175fa2c7e50>: None}>

<ConditionValue {<Request() object at 0x175fd503460>: None}>

<ConditionValue {<Request() object at 0x175fd503790>: None}>

aula Curso 1

aula Curso 2

aula Curso 3

aula Curso 4

aula Curso 5

aula Curso 6

<ConditionValue {<Request() object at 0x175fd5c50a0>: None}>

In [ ]:

1