

# Programa ejemplo juguetería.

## Definición de parámetros.

Lo primero que tenemos que hacer es **definir las variables de estado**. Como nos interesa conocer el número de clientes que habrá en la juguetería a una hora determinada, solo tenemos que definir una variable en este ejemplo:

**N = número de clientes en la juguetería.**

En segundo lugar, definimos los posibles eventos. En este caso, como solo nos interesa conocer N, **podemos definir dos procesos en los que el estado del sistema se vería modificado:**

**Llega un cliente:  $N = N + 1$**

**Se va un cliente:  $N = N - 1$**

Por último, definimos todas aquellas variables que sean necesarias para llevar a cabo la simulación, ya sea el reloj o algunos contadores.

- T: Reloj de simulación.
- DL: Tiempo entre llegadas, cuya distribución F es conocida. Los tiempos los da el enunciado porque todavía no sabemos calcularlos.
- DS: Tiempos de servicio, cuya distribución G es conocida. Los tiempos los da el enunciado porque todavía no sabemos calcularlos.
- Defino TL y TS como variables que representan el tiempo de próxima llegada y próximo servicio para poder ir avanzando el reloj.

Es importante tener en cuenta que en este problema suponemos que hay un único dependiente y que en el instante inicial o de apertura de tienda, el número de clientes en la juguetería es cero:  $N = 0$  con  $T = 0$ .

## Código.

```
% Defino las variables de entrada

T = 0;
N = 0;

DL = [3 2 5 1 2 6 6 2 8];
DS = [4 1 4 1 3 2 3 5];

TS = inf; % No hay clientes, no pueden ser atendidos
TL = DL(1); % Genero el tiempo en el que llegará el primer cliente

% Necesito dos contadores para saber el orden de llegada y de servicio
% en los que se encuentran la simulación

cl = 2; % El prox cliente que entra es el segundo
cs = 1; % El prox cliente que sale es el primero
```

```
Nclientes = prog_principal(T,N,DL,DS,TL,TS,cl,cs)
```

```
reloj = 3
v1 = 1
reloj = 5
v1 = 2
reloj = 7
v1 = 1
reloj = 8
v1 = 0
reloj = 10
v1 = 1
reloj = 11
v1 = 2
reloj = 13
v1 = 3
reloj = 14
v1 = 2
reloj = 15
v1 = 1
reloj = 18
v1 = 0
reloj = 19
v1 = 1
reloj = 21
v1 = 0
reloj = 25
v1 = 1
reloj = 27
v1 = 2
reloj = 28
v1 = 1
reloj = 33
v1 = 0
Nclientes = 0
Nclientes = 0
```

```
function Nclientes = prog_principal(reloj,v1,llegadas,salidas,pllegada,psalida,cllegada,csalida)

% REGLA DE PARADA
% Lo haremos con un bucle while en esta ocasión
while reloj < 33 % Pongo este criterio para comprobar con respecto a la traza

    % RUTINA DE TIEMPO
    % Avanzo el reloj de simulación al próximo evento
    reloj = min(pllegada,psalida);

    % RUTINAS DE ESTADO
    % Compruebo si ese evento es una llegada o una salida
    if pllegada < psalida
        [reloj,v1,llegadas,salidas,pllegada,psalida,cllegada,csalida] = rutina_llegada(reloj,v1,llegadas,salidas,pllegada,psalida,cllegada,csalida);
    else
        [reloj,v1,llegadas,salidas,pllegada,psalida,cllegada,csalida] = rutina_salida(reloj,v1,llegadas,salidas,pllegada,psalida,cllegada,csalida);
    end

    reloj
    v1
```

```

end

Nclientes = v1

end

% SUBROUTINA DE LLEGADA

function [reloj,v1,llegadas,salidas,pllegada,psalida,cllegada,csalida] = rutina_llegada(reloj,v1,llegadas,salidas,pllegada,psalida,cllegada,csalida)

    v1 = v1 + 1; % Un nuevo cliente en la tienda

    % ¿Qué pasa si hay clientes esperando a ser atendidos?

    if v1 <= 1 % No hay clientes antes

        psalida = reloj + salidas(csalida); % Genero el tiempo que tardará en ser atendido
        csalida = csalida + 1; % Avanzo el contador de salidas

    end

    % Genero los datos para la próxima llegada
    pllegada = reloj + llegadas(cllegada); % Genero la hora de la próxima llegada
    cllegada = cllegada + 1; % Avanzo el contador

end

% SUBROUTINA DE SALIDA

function [reloj,v1,llegadas,salidas,pllegada,psalida,cllegada,csalida] = rutina_salida(reloj,v1,llegadas,salidas,pllegada,psalida,cllegada,csalida)

    v1 = v1 - 1; % Sale un cliente de la tienda

    % ¿Qué pasa si no hay clientes en la tienda?

    if v1 > 0 % Sí hay clientes en la tienda

        psalida = reloj + salidas(csalida); % Genero el tiempo que tardará en ser atendido el s
        csalida = csalida + 1; % Avanzo el contador de salidas

    else % No hay clientes en la tienda

        psalida = inf;

    end

end
end

```

