

El algoritmo siguiente es la modificación del original (el que aparece en los apuntes del curso) y trata el caso de las interrupciones de servicio. Los clientes que estaban siendo atendidos salen más tarde ya que deben considerar la interrupción. Los clientes que llegan cuando el servicio esta interrumpido y hay capacidad para atenderlos, deben esperar a que se reestablezca el servicio (termine la interrupción).

```
leer archivos
inicializar el manejador de eventos
inicializar variables - Entre otras hacer Tiempo_simulado = 0
obtener el próximo evento a simular y actualizar Tiempo_simulado
mientras Tiempo_simulado ≤ Tiempo_de_simulación
    para el nodo en donde ocurre el evento haga
        si el evento es una interrupción entonces
            bandera_de_interrupción = TRUE
            calcula tiempo_de_interrupcion
            genera fin_de_interrupcion (* nuevo evento *)
            recorre la LEP (* evento por evento *)
                si el evento es una salida del nodo donde ocurre evento la interrupción
                    sacar el evento de la LEP
                    agregar al tiempo el tiempo_de_interrupcion
                    colocar en la pila
            fin recorre la LEP
        mientras la pila no este vacia
            (* todos los eventos que tuvieron cambio de tiempo de salida, se vuelven a insertar en la LEP *)
            sacar evento de la pila
            insertar en la LEP
        fin del mientras
    de lo contrario si el evento es fin de interrupcion
        bandera_de_interrupción = FALSE
        genera próxima interrupcion
        mientras (cola > 0) Y (utilización < capacidad)
            (* dejamos pasar a los clientes que se quedaron en cola, habiendo capacidad, porque estaba interrumpido el servicio *)
            cola = cola - 1
            utilización = utilización + 1
            generar salida del cliente
        fin del mientras
    de lo contrario si el evento es una llegada entonces
        si (utilización = capacidad) O (bandera_de_interrupción = TRUE) entonces
            (* si hay interrupción hay que esperar en cola a que termine *)
            cola = cola + 1
        de lo contrario
            genera la salida del cliente
            utilización = utilización + 1
        si la llegada fue externa entonces
            genera la próxima llegada externa
    de lo contrario (* salida *)
        si hay nodos sucesores entonces
            determinar a que nodo ira el cliente
            generar dicha llegada
        si hay clientes en cola entonces
            cola = cola - 1
            generar salida del cliente
        de lo contrario
            utilización = utilización - 1
        obtener el próximo evento a simular y actualizar Tiempo_simulado
    fin del mientras
```