

Los Fumadores

Este programa utiliza 5 variables globales:

- *papel*: Semáforo utilizado para controlar la existencia del papel en el estanco.
- *tabaco*: Semáforo utilizado para controlar la existencia del tabaco en el estanco.
- *cerillas*: Semáforo utilizado para controlar la existencia del cerillas en el estanco.
- *estanquero*: Semáforo utilizado para controlar la producción del estanco.
- *mutex*: Semáforo para cumplir la exclusión mutua.

Utilizaremos 2 funciones en este programa:

- *retraso_aleatorio*: suspende el programa durante un tiempo aleatorio.
- *fumar*: Ejecuta la acción de fumar.

```
23
24 void retraso_aleatorio( const float smin, const float smax )
25 {
26     static bool primera = true ;
27     if ( primera ) // si es la primera vez:
28     { srand(time(NULL)); // inicializar la semilla del generador
29       primera = false ; // no repetir la inicialización
30     }
31     // calcular un número de segundos aleatorio, entre {\ttbf smin} y {\ttbf smax}
32     const float tsec = smin+(smax-smin)*((float)random())/((float)RAND_MAX);
33     // dormir la hebra (los segundos se pasan a microsegundos, multiplicándos por 1 millón)
34     usleep( (useconds_t) (tsec*1000000.0) );
35 }
36

41
42 void fumar( int num_fumador )
43 {
44     cout << "\t\t\t\tFumador número " << num_fumador << ": comienza a fumar." << endl << flush ;
45     retraso_aleatorio( 0.2, 0.8 );
46     cout << "\t\t\t\tFumador número " << num_fumador << ": termina de fumar." << endl << flush ;
47 }
...
```

Este programa se base en la ejecución de 3 hebras:

- *funcion_fumador*: Es la función que ejecuta la hebra “fumadora”.
- *funcion_estanquero*: Es la función que ejecuta la hebra “estanquero”.
- *main*: Es la hebra encargada de crear al resto de hebras descritas.

```
49
50 void *funcion_fumador(void *id_fumador) {
51     unsigned long fumador = (unsigned long) id_fumador;
52
53     if(fumador==0) {
54         while(true) {
55             sem_wait(&cerillas);
56             sem_wait(&mutex);
57             fumar(0);
58             sem_post(&mutex);
59             sem_post(&estanquero);
60         }
61     }else if(fumador==1) {
62         while(true) {
63             sem_wait(&tabaco);
64             sem_wait(&mutex);
65             fumar(1);
66             sem_post(&mutex);
67             sem_post(&estanquero);
68         }
69     }else{
70         while(true) {
71             sem_wait(&papel);
72             sem_wait(&mutex);
73             fumar(2);
74             sem_post(&mutex);
75             sem_post(&estanquero);
76         }
77     }
78 }
79
```

```

79
80 void *funcion_estanquero(void *id_estanquero){
81     int ingrediente;
82     while(true) {
83         sem_wait(&estanquero);
84         ingrediente=(int)random()%3;
85         if(ingrediente==0) {
86             sem_wait(&mutex);
87             cout << "El estanquero produce papel" << endl << flush;
88             sem_post(&mutex);
89             sem_post(&papel);
90         }else if(ingrediente==1) {
91             sem_wait(&mutex);
92             cout << "El estanquero produce tabaco" << endl << flush;
93             sem_post(&mutex);
94             sem_post(&tabaco);
95         }else {
96             sem_wait(&mutex);
97             cout << "El estanquero produce cerillas" << endl << flush;
98             sem_post(&mutex);
99             sem_post(&cerillas);
100         }
101     }
102 }
103
104 int main() {
105
106     srand( time(NULL) );
107
108     pthread_t h_fumador,h_estanquero;
109
110     sem_init(&estanquero,0,1);
111     sem_init(&papel,0,0);
112     sem_init(&cerillas,0,0);
113     sem_init(&tabaco,0,0);
114     sem_init(&mutex,0,1);
115
116
117     for(unsigned long i =0; i<3; i++)
118         pthread_create(&h_fumador,NULL,funcion_fumador,(void *) i);
119
120     pthread_create(&h_estanquero,NULL,funcion_estanquero,NULL);
121
122     pthread_exit(NULL);
123
124
125
126     return 0 ;
127 }

```

En primer lugar declaramos las cuatro hebras adicionales que vamos a utilizar (*h_fumador* y *h_estanquero*) .

A continuación debemos inicializar los semáforos con los valores correctos:

- *estanquero* lo inicializamos a 1 para que el estanquero pueda proporcionar el ingrediente.
- *papel* lo inicializamos a 0 porque el fumador no tiene inicialmente ningún item para fumar.
- *tabaco* lo inicializamos a 0 porque el fumador no tiene inicialmente ningún item para fumar.
- *cerillas* lo inicializamos a 0 porque el fumador no tiene inicialmente ningún item para fumar.
- *mutex* es un semáforo para cumplir la propiedad de exclusión mutua, por defecto se inicializa a 1.

Una vez hecho todo esto, pasaremos a la creación de las hebras asignandole a cada una, una función que llevar a cabo. Tres de ellas ejecutarán la *funcion_fumadores* la restante ejecutará *funcion_estanquero*.

La *h_estanquero* produce un ingrediente aleatorio (tabaco, cerillas o papel). Dependiendo del ingrediente producido da paso a la *h_fumadora* que necesite ese ingrediente para fumar y se bloquea esperando nuevas peticiones.

La *h_fumadora* consume el ingrediente que proporciona la *h_estanquero* y le envía una petición al mismo para que produzca otro nuevo ingrediente.