

SISTEMAS OPERATIVOS

Solución PED1 (Noviembre 2015)

Solución Ejercicio 1

- I) Dentro del código de un proceso pueden existir varias secciones críticas. En conclusión la afirmación es **FALSA**.
- II) Esta afirmación es **VERDADERA** ya que la creación de un hilo nuevo dentro de un proceso ya existente requiere de menos tiempo que la creación de un nuevo proceso. En algunos sistemas crear un hilo es varios órdenes de magnitud más rápido que crear un proceso. Asimismo el tiempo necesario para finalizar un hilo es menor que el tiempo necesario para finalizar un proceso. Además, un cambio de proceso requiere más tiempo que un cambio de hilo dentro de un mismo proceso.
- III) El sistema operativo Mac OS es un sistema multitarea además hasta su versión 10 era un sistema monousuario y a partir de su versión 10 (más conocida como Mac OS X o OS X) es un sistema multiusuario. Como en el enunciado se indica Mac OS y no Mac OS X entonces la afirmación es **FALSA** ya que Mac OS es monousuario.
- IV) Esta afirmación es **FALSA** ya que la definición dada en la afirmación para el tiempo de espera corresponde en realidad al *tiempo de respuesta*, el tiempo de espera se define como la suma de los tiempos que un proceso pasa esperando en las diferentes colas del sistema por la obtención de recursos.

Solución Ejercicio 2

- a) El diagrama de uso del procesador se muestra en la Figura 1. Por otra parte en la Tabla 1 se muestra para cada proceso en orden de planificación su tiempo de finalización y su tiempo de espera. Recuerdese que el tiempo de espera se calcula como el tiempo de finalización menos el tiempo de servicio.

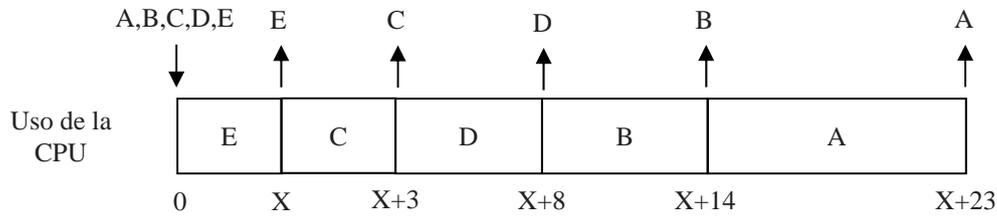


Figura 1

Proceso	Tiempo de finalización (ut)	Tiempo de espera (ut)
E	X	0
C	$X + 3$	X
D	$X + 8$	$X + 3$
B	$X + 14$	$X + 8$
A	$X + 23$	$X + 14$

Tabla 1

El tiempo de espera medio se calcula como la suma de los tiempos de espera de cada proceso partido por el número de procesos:

$$\frac{0 + X + (X + 3) + (X + 8) + (X + 14)}{5} = \frac{4X + 25}{5}$$

b) El diagrama de uso del procesador se muestra en la Figura 2. En la Tabla 2 se muestra para cada proceso en orden de planificación su tiempo de finalización y su tiempo de espera.

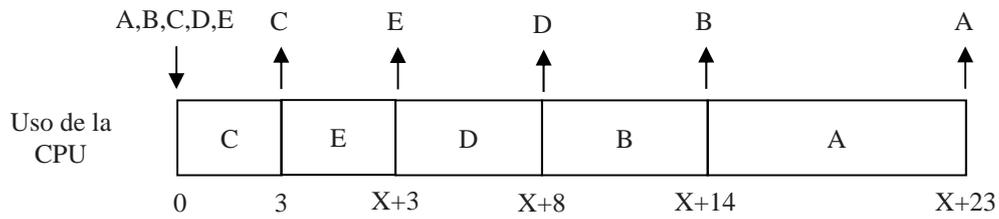


Figura 2

Proceso	Tiempo de finalización (ut)	Tiempo de espera (ut)
C	3	0
E	X + 3	3
D	X + 8	X + 3
B	X + 14	X + 8
A	X + 23	X + 14

Tabla 2

El tiempo de espera medio se calcula como la suma de los tiempos de espera de cada proceso partido por el número de procesos:

$$\frac{0 + 3 + (X + 3) + (X + 8) + (X + 14)}{5} = \frac{3X + 28}{5}$$

c) El diagrama de uso del procesador se muestra en la Figura 3. En la Tabla 3 se muestra para cada proceso en orden de planificación su tiempo de finalización y su tiempo de espera.

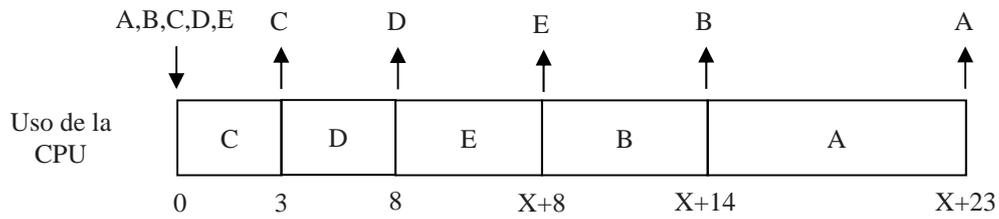


Figura 3

Proceso	Tiempo de finalización (ut)	Tiempo de espera (ut)
C	3	0
D	8	3
E	X + 8	8
B	X + 14	X + 8
A	X + 23	X + 14

Tabla 3

El tiempo de espera medio se calcula como la suma de los tiempos de espera de cada proceso partido por el número de procesos:

$$\frac{0 + 3 + 8 + (X + 8) + (X + 14)}{5} = \frac{2X + 33}{5}$$

d) El diagrama de uso del procesador se muestra en la Figura 4. En la Tabla 4 se muestra para cada proceso en orden de planificación su tiempo de finalización y su tiempo de espera.

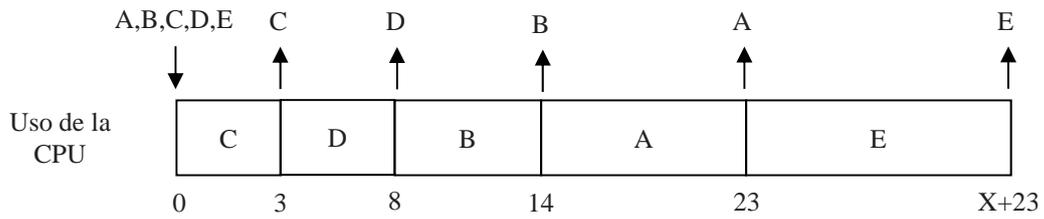


Figura 4

Proceso	Tiempo de finalización (ut)	Tiempo de espera (ut)
C	3	0
D	8	3
B	14	8
A	23	14
E	X + 23	23

Tabla 4

El tiempo de espera medio se calcula como la suma de los tiempos de espera de cada proceso partido por el número de procesos:

$$\frac{0 + 3 + 8 + 14 + 23}{5} = \frac{48}{5}$$

Solución Ejercicio 3

En la Figura 5 se muestra una posible solución para este ejercicio. Esta solución utiliza las siguientes variables y semáforos:

- `contadorPA`. Variable global de tipo entero para llevar la cuenta del número de peticiones de aterrizaje.
- `contadorEDA`. Variable global de tipo entero para llevar la cuenta del número de aviones esperando para despegar por existir aterrizajes.
- S1. Semáforo binario para garantizar la exclusión mutua en el uso de la variable `contadorPA`.
- S2. Semáforo binario para garantizar la exclusión mutua en el uso de la variable `contadorEDA`.
- S3. Semáforo binario para garantizar la exclusión mutua en el uso de la pista de aterrizaje.
- S4. Semáforo binario para avisar que no existen peticiones de aterrizaje a los aviones que desean despegar.

```

int contadorEDA, contadorPA; /* Definición variables */
semáforo binario S1, S2, S3, S4; /* Definición semáforos */

void avion_aterriza() /* Código proceso avión_aterriza */
{
    wait_sem(S1);
    contadorPA=contadorPA+1;
    signal_sem(S1);

    wait_sem(S3); /* Esperar si otro avión está usando la pista */
    aterrizar();
    signal_sem(S3);

    wait_sem(S1);
    contadorPA=contadorPA-1;
    if (contadorPA==0) /* Avisar a los aviones que esperan para despegar */
    {
        wait_sem(S2);
        while(contadorEDA>0){
            signal_sem(S4);
            contadorEDA=contadorEDA-1;
        }
        signal_sem(S2);
        signal_sem(S1);
    }
    else signal_sem(S1);
}

void avion_despega() /* Código proceso avión_despega */
{
    wait_sem(S1);
    if (contadorPA>0) /* Esperar si hay peticiones de aterrizaje */
    {
        wait_sem(S2);
        contadorEDA=contadorEDA+1;
        signal_sem(S2);
        signal_sem(S1);

        wait_sem(S4);
    }
    else signal_sem(S1);

    wait_sem(S3); /* Esperar si otro avión está usando la pista */
    despegar();
    signal_sem(S3);
}

main() /* Inicialización de semáforos y ejecución concurrente */
{
    init_sem(S1,1);
    init_sem(S2,1);
    init_sem(S3,1);
    init_sem(S4,0);
    ejecución_concurrente(avion_aterriza,avion_aterriza,...,avion_despega,avion_despega,...);
}

```

Figura 5