los algoritmos que se utilizan para la ordenación de procesos y cómo aplican las preferencias para éstos cada uno de los distintos tipos de planificadores.

Por qué la necesidad de estos?

Estos métodos nacieron por la necesidad de poder ordenar los procesos para ganar eficiencia a la hora de tratar con ellos, es decir, son los encargados de ordenar y dirigir los procesos para asegurar que ninguno de ellos monopolice el uso de la CPU.

**Conceptos básicos**.

Antes de ver algunos de los algoritmos más utilizados vamos a dar a conocer algunos aspectos o conceptos para entenderlos mejor.

**Tiempo de espera**: El tiempo que un proceso permanece en espera en la cola de ejecución.

**Tiempo de retorno**: Tiempo que va desde que se lanza un proceso hasta que finaliza.

**Tiempo de respuesta**: Por último éste se define a el tiempo que un proceso bloqueado tarda en entrar en ejecución.

**Uso de CPU**: Porcentaje de tiempo que la CPU está ocupada.

**Productividad:** Número de procesos realizados en una unidad de tiempo.

Y por último dos tipos de algoritmos:

**Apropiativo**: También conocido como expulsivo o expropiativo, este tipo de algoritmo nos permite la expulsión de procesos para ejecutar un nuevo proceso, poniendo en cola al anterior.

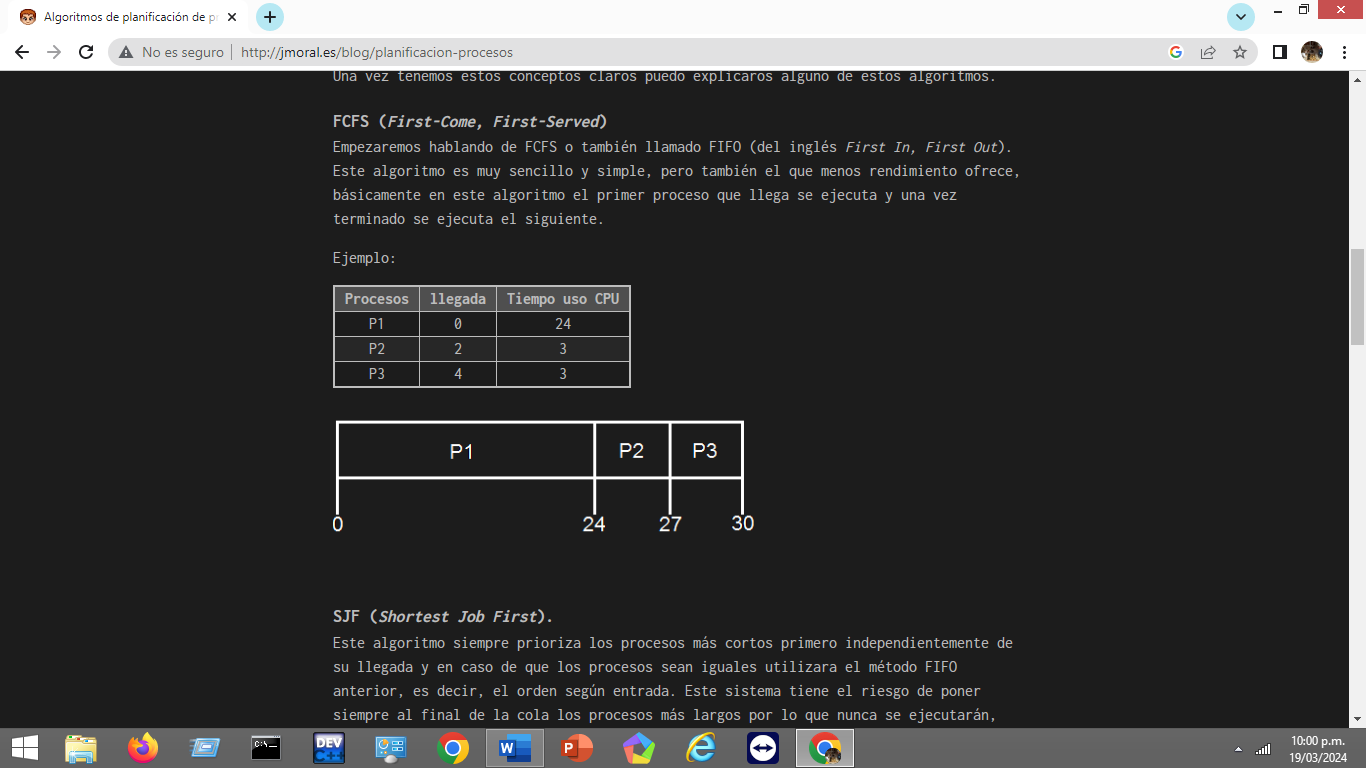
**No Apropiativo**: Este tipo no nos permite la expulsión, por lo que un proceso nuevo no entrará hasta que termine el anterior.

**Tipos de algoritmo.**

Una vez tenemos estos conceptos claros puedo explicaros alguno de estos algoritmos.

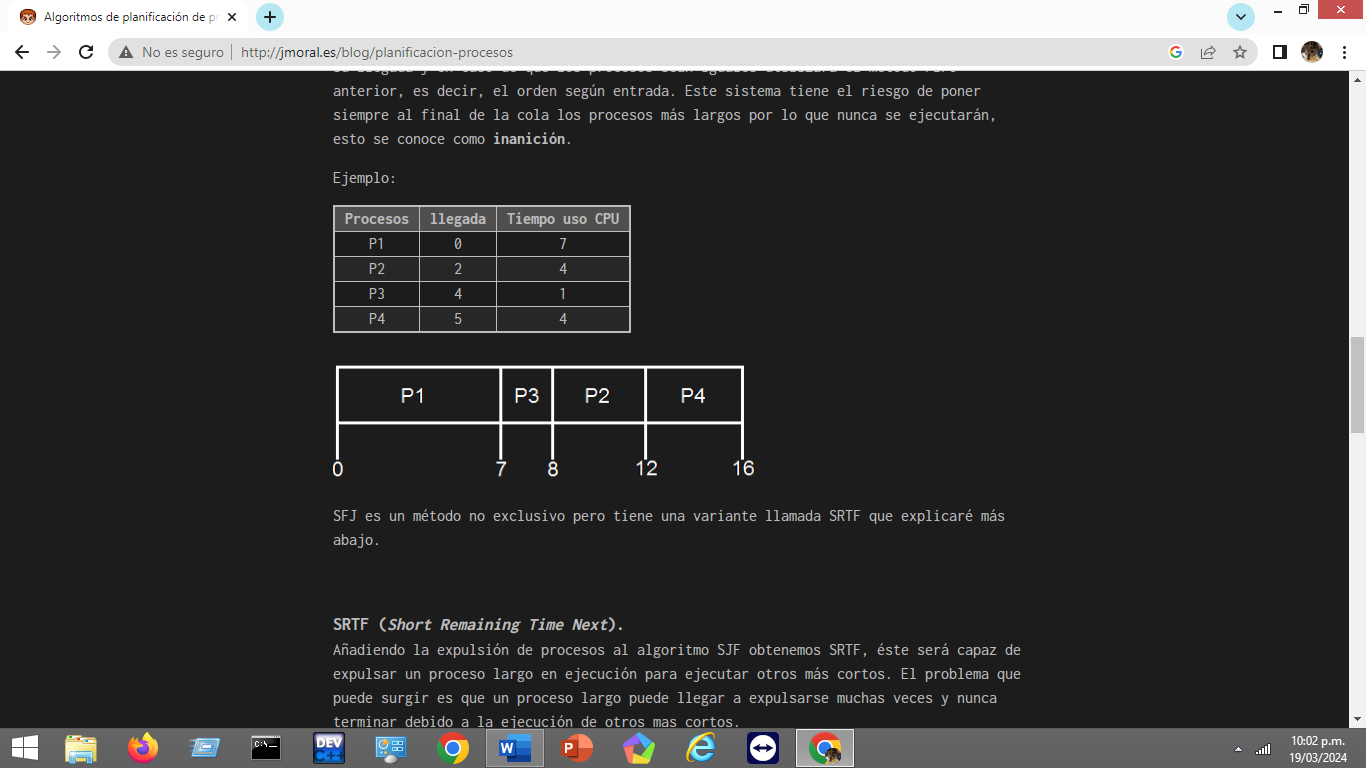
**FCFS (First-Come, First-Served)**

Empezaremos hablando de FCFS o también llamado FIFO (del inglés First In, First Out). Este algoritmo es muy sencillo y simple, pero también el que menos rendimiento ofrece, básicamente en este algoritmo el primer proceso que llega se ejecuta y una vez terminado se ejecuta el siguiente.



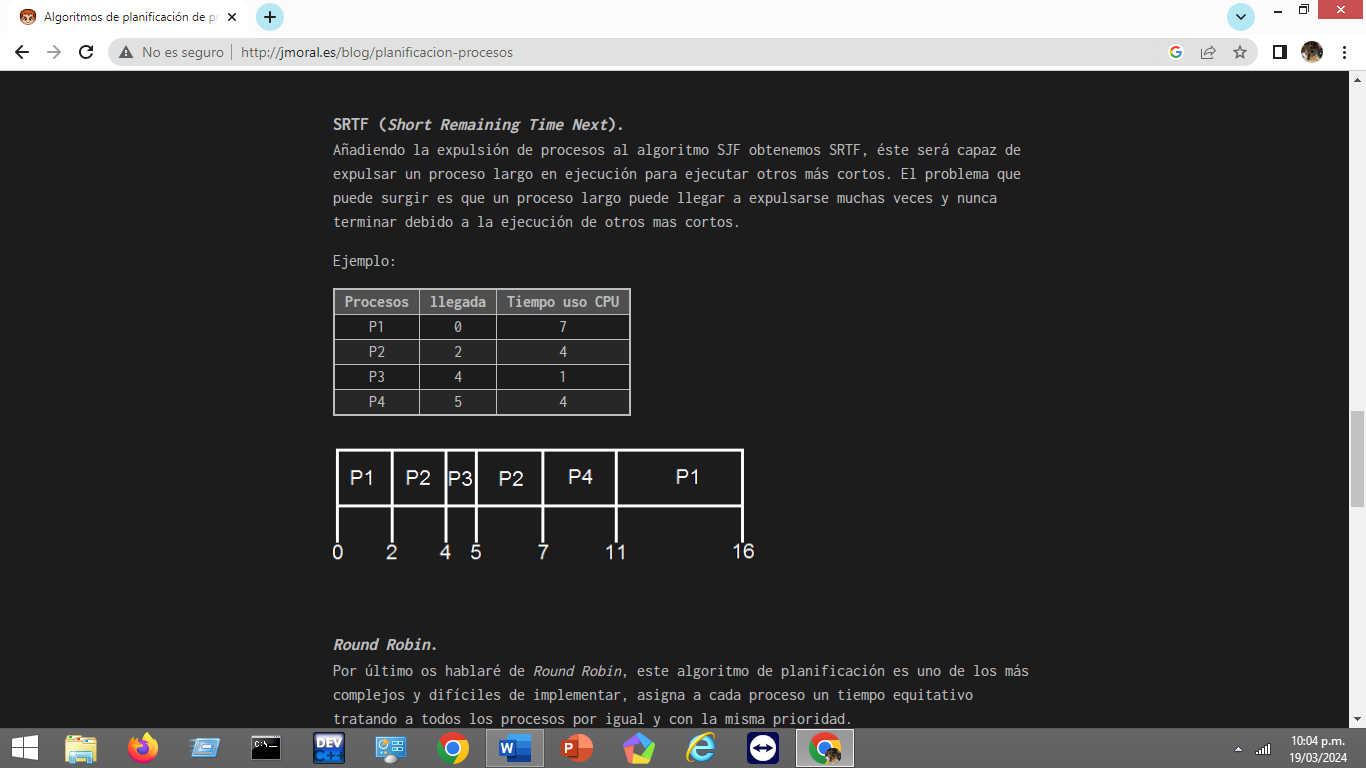
**SJF (Shortest Job First).**

Este algoritmo siempre prioriza los procesos más cortos primero independientemente de su llegada y en caso de que los procesos sean iguales utilizara el método FIFO anterior, es decir, el orden según entrada. Este sistema tiene el riesgo de poner siempre al final de la cola los procesos más largos por lo que nunca se ejecutarán, esto se conoce como inanición.



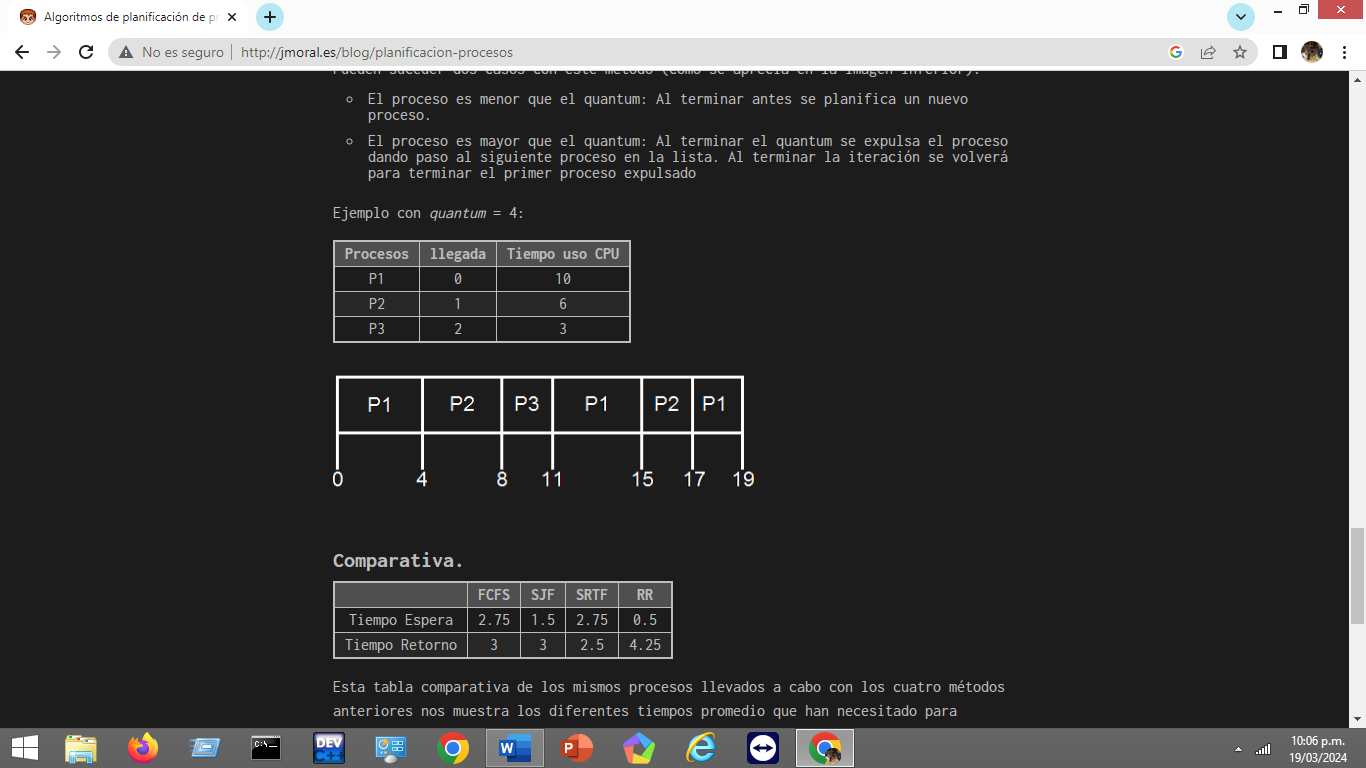
**SRTF (Short Remaining Time Next).**

Añadiendo la expulsión de procesos al algoritmo SJF obtenemos SRTF, éste será capaz de expulsar un proceso largo en ejecución para ejecutar otros más cortos. El problema que puede surgir es que un proceso largo puede llegar a expulsarse muchas veces y nunca terminar debido a la ejecución de otros mas cortos.



**Round Robin**.

Por último os hablaré de Round Robin, este algoritmo de planificación es uno de los más complejos y difíciles de implementar, asigna a cada proceso un tiempo equitativo tratando a todos los procesos por igual y con la misma prioridad.



Este algoritmo es circular, volviendo siempre al primer proceso una vez terminado con el último, para controlar este método a cada proceso se le asigna un intervalo de tiempo llamado quantum o cuanto (para definirlo se utiliza esta regla, el 80% de los procesos tienen que durar menos tiempo que el quantum definido).

Pueden suceder dos casos con este método (como se aprecia en la imagen inferior):

El proceso es menor que el quantum: Al terminar antes se planifica un nuevo proceso.

El proceso es mayor que el quantum: Al terminar el quantum se expulsa el proceso dando paso al siguiente proceso en la lista. Al terminar la iteración se volverá para terminar el primer proceso expulsado