Tabla de contenido

[Errores que hemos tenido hasta ahora 2](#_Toc138269534)

[Funciones del programa: 3](#_Toc138269535)

[Funciones de lectura de datos (Línea 5) 3](#_Toc138269536)

[Funciones de Informe (Línea 322) 3](#_Toc138269537)

[Funciones de místicos (Línea 353) 3](#_Toc138269538)

[Funciones de Inicialización (Línea 409) 3](#_Toc138269539)

[Funciones de Introducción (Línea 569) 3](#_Toc138269540)

[Funciones de Opciones (Línea 866) 3](#_Toc138269541)

[Funciones de Datos (Línea 843) 4](#_Toc138269542)

[Funciones de datos de memoria (Línea 1446) 4](#_Toc138269543)

[Funciones de datos por teclado (Línea 1657) 4](#_Toc138269544)

[Funciones de datos por archivos (Línea 1931) 4](#_Toc138269545)

[Funciones de datos randoms (Línea 2651) 5](#_Toc138269546)

[Funciones de informes (Línea 3579) 5](#_Toc138269547)

[Funciones de Ejecución (Línea 3722) 6](#_Toc138269548)

[Funciones de pantalla (Línea 4453) 6](#_Toc138269549)

[Protocolo SRPT 7](#_Toc138269550)

[Ejercicio hecho a mano 8](#_Toc138269551)

[Medicaciones Realizadas 17](#_Toc138269552)

# Errores que hemos tenido hasta ahora

Problemas con la ejecución al necesitar instalar el paquete BC que no estaba instalado

sudo apt-get install bc

# Funciones del programa:

## Funciones de lectura de datos (Línea 5)

* Leer() -> Lee la variable dada en raw, para que lo interprete como texto
* Leer\_numero() -> Lee el número que le pasamos y comprueba que sea un número entre 0 y el valor máximo que permite bash
* Leer\_numero\_entre() -> Lee un número que debe estar entre 2 valores. (Número máximo opcional)
* Leer\_nombre\_archivo() -> Lee un nombre de archivo válido según los criterios que hemos puesto
* Preguntar\_segundos() -> Muestra una pantalla de pregunta genérica con los parámetros dados para
* Preguntar() -> Muestra una pantalla de pregunta genérica con los parámetros dados
* Preguntar\_si\_no() -> Pregunta de respuesta sí o no. No se guarda en informes

### Funciones de Informe (Línea 322)

* Informar\_plano() -> Añade cadena a la cadena de informe plano. Se usa como si de un printf o salto de linea
* Informar\_color() -> Añade cadena a la cadena de informe a color Se usa como si de un printf se tratara.
* Guardar\_informes() -> Guarda las cadenas de informe a sus archivos respectivos y las vacía.

### Funciones de místicos (Línea 353)

* Cabecera() -> Muestra una cabecera general
* Aleatorio\_entre() -> Crea un número pseudoaleatorio entre un máximo y un mínimo y lo asigna a la variable.
* Pausa\_tecla() -> Espera a que se pulse una tecla para continuar el programa
* Barra\_loading() -> Muestra una barra tan ancha como la terminal con la proporción $1 / $2 siendo el valor actual y el total.

## Funciones de Inicialización (Línea 409)

* Init\_globales() -> Establece las variables globales.
* Init\_colores() -> Establece las variables de color.
* Init() -> Se inicializan variables globales.

## Funciones de Introducción (Línea 569)

* Intro\_cabecera\_inicio() -> Muestra la cabecera con datos relevantes.
* Intro\_cabecera\_tamano() -> Muestra la cabecera con aviso sobre el tamaño de la terminal.
* Intro() -> Se muestran las cabeceras.

## Funciones de Opciones (Línea 866)

Oopciones\_informes() -> Da a elegir si se desea cambiar los informes por defecto.

* Opciones\_menu\_ayuda() -> Muestra la ayuda del fichero de ayuda si este existe.
* Opciones\_menu() -> Elige si mostrar la ayuda o ejecutar el algoritmo.
* Opciones() -> Función principal de opciones.

## Funciones de Datos (Línea 843)

* datos\_pregunta\_guardar\_rangos() -> Pregunta si se desean guardar los rangos.
* datos\_rango\_guardar()-> Guardar los datos a archivo.
* datos\_pregunta\_guardar()-> Pregunta si se desean guardar los procesos.
* datos\_guardar() -> Guardar los datos a archivo.
* generar\_nombre\_proceso() -> Crea los nombres de los procesos ej 1 -> P01.
* datos\_tabla\_procesos() -> Muestra una tabla con todos los procesos introducidos hasta el momento.
* datos\_almacena\_marcos() ->
* datos\_obtiene\_marcos() ->
* datos\_ordenar\_llegada() ->Ordena los procesos según llegada en la lista de llegada.

### Funciones de datos de memoria (Línea 1446)

* datos\_memoria\_tabla() -> Muestra una tabla con las características de la memoria según se van dando.
* datos\_memoria\_tamaño() -> Introducir número de direcciones de la memoria.
* datos\_memoria\_tamaño\_pagina() -> Introducir tamaño de página.
* datos\_memoria\_numero\_marcos() -> Calcular número de marcos.
* datos\_memoria\_mNur() -> Introducir mínimo para que se produzca reubicación (Solo NC - R).
* datos\_memoria() -> Introducir características de la memoria.

### Funciones de datos por teclado (Línea 1657)

* datos\_teclado\_llegada() -> Pide el tiempo de llegada del proceso.
* datos\_teclado\_nm() -> Pide el mínimo estructural del proceso.
* datos\_teclado\_direcciones() -> Va pidiendo las direcciones del proceso.
* datos\_teclado() -> Introducir los datos por teclado.

### Funciones de datos por archivos (Línea 1931)

* datos\_archivo\_comprobar() -> Comprueba que la carpeta existe y que hay archivos dentro. También crea la lista con los archivos que hay dentro.
* datos\_archivo\_rangos\_comprobar() -> Comprueba que la carpeta existe y que hay archivos dentro. También crea la lista con los archivos que hay dentro.
* datos\_archivo\_seleccionar() -> Muestra una lista con todos los archivos de la que se puede seleccionar el que se quiera.
* datos\_archivo\_informes() -> Añade a los informes el archivo que se va a usar.
* datos\_archivo\_leer() -> Leer los datos del archivo seleccionado.
* datos\_archivo\_rangos\_leer() -> Leer los datos del archivo de rangos seleccionado.
* datos\_archivo\_ultima\_ejecucion() -> Introducir los datos de la última ejecución.
* datos\_archivo\_ultima\_ejecucion\_random() -> Introducir los datos de la última ejecución.
* datos\_archivo() -> Introducir los datos mediante archivo.
* datos\_archivo\_rangos() -> Introducir los datos mediante archivo de rangos.

### Funciones de datos randoms (Línea 2651)

* datos\_random\_tabla1() -> Muestra los parámetros para la generación de datos de memoria.
* datos\_random\_informes1() -> Añade la tabla con los parámetros a los informes de datos de memoria.
* datos\_random\_procesos() -> Introducir número de procesos a crear.
* datos\_random\_memoria() -> Introducir tamaños de memoria.
* datos\_random\_pagina() -> Introducir tamaño de páginas.
* datos\_random\_marcos() -> Introducir el número de marcos.
* datos\_random\_reubicacion() -> Introducir mínimo para la reubicación.
* datos\_random\_llegada() -> Introducir tiempos de llegada.
* datos\_random\_ejecucion() -> Introducir tiempos de ejecución.
* datos\_random\_nm() -> Introducir mínimos estructurales.
* datos\_random\_direcciones() -> Introducir rango de direcciones.
* datos\_random() -> Generar los procesos de forma pseudo-aleatoria.

### Funciones de informes (Línea 3579)

* datos\_informar() -> Añade informes sobre características de la memoria y los procesos.
* datos() -> Se introducen datos sobre memoria y procesos.

## Funciones de Ejecución (Línea 3722)

* ej\_ejecutar\_tesp\_tret() -> Calcular tiempo de espera y de ejecución para los procesost
* ej\_ejecutar\_fin\_ejecutar() -> Finalizar la ejecución del proceso
* ej\_ejecutar\_comprobar\_reubicacion() -> Comprobar si se cumplen las condiciones para que se produzca reubicación
* ej\_ejecutar\_reubicar() -> Reubicar la memoria
* ej\_ejecutar\_llegada() -> Atender la llegada de procesos
* ej\_ejecutar\_memoria\_proceso() -> Introducir procesos que han llegado a memoria si se puede
* ej\_ejecutar\_empezar\_ejecucion() -> Meter proceso al procesador
* ej\_ejecutar\_memoria\_pagina() -> Introducir siguiente página del proceso a memoria
* ej\_calcular\_marco\_siguiente() -> Encuentra cual va a ser el siguiente marco en utilizar en caso de que se produzca fallo en la siguiente página
* ej\_ejecutar\_guardar\_fallos() -> Guardar el estado de la memoria en este momento para luego mostrar el resumen con los fallos. No está directamente relacionado con la ejecución. Es solo para la pantalla.
* ej\_ejecutar() -> Llegada de procesos, ejecución, introducción a memoria...

### Funciones de pantalla (Línea 4453)

* ej\_pantalla\_cabecera() -> Mostrar una cabecera con información sobre el algoritmo y sobre la memoria
* ej\_pantalla\_tiempo() -> Mostrar el tiempo actual
* ej\_pantalla\_llegada() -> Mostrar información sobre la llegada de procesos
* ej\_pantalla\_tabla() -> Mostrar tabla con los procesos y sus datos
* ej\_pantalla\_reubicacion() -> Mostrar el cambio de memoria que ha habido en la reubicación
* ej\_pantalla\_media\_tiempos() -> Mostrar media de Tesp y de Tret
* ej\_pantalla\_fin\_fallos() -> Mostrar un resumen con los fallos de página que ha habido durante la ejecución
* ej\_pantalla\_fin() -> Mostrar el proceso que ha finalizado su ejecución
* ej\_pantalla\_entrada() -> Mostrar información sobre la entrada de procesos en memoria
* ej\_pantalla\_cola() -> Mostrar cola de ejecución
* ej\_pantalla\_inicio() -> Mostrar el proceso que ha iniciado su ejecución
* ej\_pantalla\_linea\_memoria\_grande() -> Muestra la linea de memoria grande
* ej\_pantalla\_linea\_memoria\_pequena() -> Muestra la linea de memoria pequeña.
* ej\_pantalla\_linea\_tiempo() -> Mostrar la linea temporal
* ej\_pantalla() -> Muestra la pantalla con la información de los eventos que han ocurrido
* ej\_limpiar\_eventos() -> Resetea las variables de evento para que no se vuelvan a mostrar
* ej\_resumen() -> Muestra la pantalla con la información de los eventos que han ocurrido
* ej() -> Ejecucion del algoritmo y llamadas a funciones
* main() -> Función principal

# Protocolo SRPT

Si llega un proceso más corto que lo que queda a uno para ejecutarse, le expulsa

y se ejecuta el más corto.

• Siempre está en ejecución el proceso que menos CPU necesite

• Apropiativo o expulsor. Es el modo apropiativo del SJF.

• Es útil en sistemas de tiempo compartido.

• El proceso con el tiempo estimado de ejecución menor para finalizar es el

siguiente en ser ejecutado.

• Tiene mayor sobrecarga que la planificación SJF.

• Debe mantener un registro del tiempo de servicio transcurrido del proceso en

ejecución, lo que aumenta la sobrecarga.

• La apropiación de un proceso a punto de terminar por otro de menor duración

recién llegado podría significar un mayor tiempo de cambio de contexto

(administración del procesador) que el tiempo de finalización del primero.

• Al diseñarse los Sistemas Operativos se debe considerar cuidadosamente la

sobrecarga de los mecanismos de administración de recursos comparándola

con los beneficios esperados.

# Ejercicio hecho a mano

**Datos iniciales del ejemplo:**

Número de direcciones: 2000

Tamaño de página: 100

Número de marcos: 20

Mínimo para realizar reubicación: 2

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Ref. | T. Lleg | T.Ejec | Nº Marcos | Págs- Dirs. | T.Rest |
| P01 | 2 | 11 | 4 | 323-**3** 784-**7** 523-**5** 815-**8** 765-**7** 891-**8** 654-**6** 454-**4** 498-**4** 987-**9** 976-**9** | 11 |
| P02 | 4 | 8 | 4 | 323-**3** 815-**8** 765-**7** 891-**8** 654-**6** 454-**4** 498-**4** 986-**9** | 8 |
| P03 | 6 | 6 | 4 | 323-**3** 124-**1** 541-**5** 891-**8** 892-**8** | 6 |
| P04 | 8 | 2 | 4 | 123-**1** 542-**5** | 2 |

El número de marcos (5) es el tamaño de la memoria o número de direcciones (500) entre el tamaño de cada página (100).

Podemos observar en los datos iniciales que los procesos no están ordenados por el tiempo de llegada, hacerlo puede facilitarnos el trabajo a la hora de utilizar el algoritmo SRPT.

El tiempo de ejecución coincide con el número de direcciones de cada proceso.

El número de marcos o mínimo estructural es el espacio que va a ocupar dicho proceso en memoria.

**A Tiempo = 0**

En este momento, simplemente se inicializan y se muestran los valores de las mandas de memoria y de tiempo vacías ya que no hay ningún proceso con tiempo de llegada igual a 0.

**Banda de memoria:**

**Vacio**

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19

**Banda temporal:**

**Vacio**

1

0 1 2 3 4 5 6 7 8 9 10 ….

**A Tiempo = 2**

Lo primero que ocurre es:

Llegada del proceso P01, con espacio en la memoria suficiente como para poder cargarlo, por lo que entra en memoria ocupando 4 marcos de página, la página 3 entra en el marco 0 que está vacío, produciendo un fallo de paginación, quedándole a este proceso un tiempo de ejecución restante de 11 procesos.

**Banda de memoria:**

**P02**

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19

**Banda temporal:**

**P02**

3

0 1 2 3 4 5 6 7 8 9 10 ….

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| P01 | | | | | | | | | | | |
| Direcciones | 3 | 7 | 5 | 8 | 7 | 8 | 6 | 4 | 4 | 9 | 9 |
| M01 | 3 |  |  |  |  |  |  |  |  |  |  |
| M02 | 1 |  |  |  |  |  |  |  |  |  |  |
| M03 |  |  |  |  |  |  |  |  |  |  |  |
| M04 |  |  |  |  |  |  |  |  |  |  |  |
| NºFallos | F1 |  |  |  |  |  |  |  |  |  |  |
| Tiempo | 2 |  |  |  |  |  |  |  |  |  |  |

**A Tiempo = 4**

En este punto, llega el proceso P02, el cual entra a ejecutarse al atenderse el algoritmo SRPT, ya que el tiempo restante de ejecución del proceso P02 es de 8, mientras que el del P01 es de 9.

Llegada del proceso P02, al igual que en el caso anterior, entra en memoria ocupando 4 marcos de página, introduciendo la página 3 en el marco 0 que está vacío, produciendo un fallo de paginación, quedándole a este proceso un tiempo de ejecución restante de 8 procesos.

**Banda de memoria:**

**P01 P02**

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19

**Banda temporal:**

**P02 P03**

3 7 3

0 1 2 3 4 5 6 7 8 9 10 ….

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| P02 | | | | | | | | |
| Direcciones | 3 | 8 | 7 | 8 | 6 | 4 | 4 | 9 |
| M01 | 3 |  |  |  |  |  |  |  |
| M02 | **1** |  |  |  |  |  |  |  |
| M03 |  |  |  |  |  |  |  |  |
| M04 |  |  |  |  |  |  |  |  |
| NºFallos | F1 |  |  |  |  |  |  |  |
| Tiempo | 4 |  |  |  |  |  |  |  |

**A Tiempo = 6**

En este punto, llega el proceso P03, el cual entra a ejecutarse al atenderse el algoritmo SRPT, ya que el tiempo restante de ejecución del proceso P03 es de 5, mientras que el del P02 es de 6.

A la llegada del proceso P03, al igual que en el caso anterior, entra en memoria ocupando 4 marcos de página, introduciendo la página 3 en el marco 0 que está vacío, produciendo un fallo de paginación, quedándole a este proceso un tiempo de ejecución restante de 5 procesos.

**Banda de memoria:**

**P01 P02 P03**

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19

**Banda temporal:**

**P01 P02 P03**

3 7 3 8 3

0 1 2 3 4 5 6 7 8 9 10 ….

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| P03 | | | | | |
| Direcciones | 3 | 1 | 5 | 8 | 8 |
| M01 | 3 |  |  |  |  |
| M02 | **1** |  |  |  |  |
| M03 |  |  |  |  |  |
| M04 |  |  |  |  |  |
| NºFallos | F1 |  |  |  |  |
| Tiempo | 6 |  |  |  |  |

**A Tiempo = 8**

En este punto, llega el proceso P04, el cual entra a ejecutarse al atenderse el algoritmo SRPT, ya que el tiempo restante de ejecución del proceso P04 es de 2, mientras que el del P03 es de 3.

A la llegada del proceso P04, al igual que en el caso anterior, entra en memoria ocupando 4 marcos de página, introduciendo la página 1 en el marco 0 que está vacío, produciendo un fallo de paginación, quedándole a este proceso un tiempo de ejecución restante de 2 procesos.

**Banda de memoria:**

**P01 P02 P03 P04**

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19

**Banda temporal:**

**P01 P02 P03 P04**

3 7 3 8 3 1 1

0 1 2 3 4 5 6 7 8 9 10 ….

|  |  |  |
| --- | --- | --- |
| P04 | | |
| Direcciones | 1 | 5 |
| M01 | 3 |  |
| M02 | **1** |  |
| M03 |  |  |
| M04 |  |  |
| NºFallos | F1 |  |
| Tiempo | 8 |  |

**A Tiempo = 10**

En este punto, el proceso P04 finaliza su ejecución con un total 2 fallos de página.

Por otra parte, el proceso P03 retoma el estado de ejecución, introduciendo la página 5 en el marco 3 que está vacío, produciendo un fallo de paginación, quedándole a este proceso un tiempo de ejecución restante de 3 procesos.

**Banda de memoria:**

**P01 P02**

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19

**Banda temporal:**

**P01 P02 P03 P04 P03**

3 7 3 8 3 1 1 5 5

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 ….

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| P03 | | | | | |
| Direcciones | 3 | 1 | 5 | 8 | 8 |
| M01 | 3 | 3 | 3 |  |  |
| M02 | **1** | 1 | 1 |  |  |
| M03 |  | **1** | 5 |  |  |
| M04 |  |  | **1** |  |  |
| NºFallos | F1 | F2 | F3 |  |  |
| Tiempo | 6 | 7 | 10 |  |  |

**Tabla resumen de fallos de paginación del Proceso finalizado**

|  |  |  |
| --- | --- | --- |
| P04 | | |
| Direcciones | 1 | 5 |
| M01 | 1 | 1 |
| M02 | **1** | 5 |
| M03 |  | **1** |
| M04 |  |  |
| NºFallos | F1 | F2 |
| Tiempo | 8 | 9 |

**A Tiempo = 13**

En este punto, el proceso P03 finaliza su ejecución con un total 4 fallos de página.

Por otra parte, el proceso P02 retoma el estado de ejecución, introduciendo la página 7 en el marco 3 que está vacío, produciendo un fallo de paginación, quedándole a este proceso un tiempo de ejecución restante de 7 procesos.

**Banda de memoria:**

**P01 P02**

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19

**Banda temporal:**

**P01 P02 P03 P04 P03 P02**

3 7 3 8 3 1 1 5 5 8 8 7

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 ….

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| P02 | | | | | | | | |
| Direcciones | 3 | 8 | 7 | 8 | 6 | 4 | 4 | 9 |
| M01 | 3 | 3 | 3 |  |  |  |  |  |
| M02 | **1** | 8 | 8 |  |  |  |  |  |
| M03 |  | **1** | 7 |  |  |  |  |  |
| M04 |  |  | **1** |  |  |  |  |  |
| NºFallos | F1 | F2 | F3 |  |  |  |  |  |
| Tiempo | 4 | 5 | 13 |  |  |  |  |  |

**Tabla resumen de fallos de paginación del Proceso finalizado**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| P03 | | | | | |
| Direcciones | 3 | 1 | 5 | 8 | 8 |
| M01 | 3 | 3 | 3 | 3 | 3 |
| M02 | **1** | 1 | 1 | 1 | 1 |
| M03 |  | **1** | 5 | 5 | 5 |
| M04 |  |  | **1** | 8 | 8 |
| NºFallos | F1 | F2 | F3 | F4 | - |
| Tiempo | 6 | 7 | 10 | 11 | 12 |

**A Tiempo = 19**

En este punto, el proceso P02 finaliza su ejecución con un total 6 fallos de página.

Por otra parte, el proceso P01 retoma el estado de ejecución, introduciendo la página 5 en el marco 3 que está vacío, produciendo un fallo de paginación, quedándole a este proceso un tiempo de ejecución restante de 9 procesos.

**Banda de memoria:**

**P01**

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19

**Banda temporal:**

**P01 P02 P03 P04 P03 P02 P01**

3 7 3 8 3 1 1 5 5 8 8 7 8 6 4 4 9 5

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 ….

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| P01 | | | | | | | | | | | |
| Direcciones | 3 | 7 | 5 | 8 | 7 | 8 | 6 | 4 | 4 | 9 | 9 |
| M01 | 3 | 3 | 3 |  |  |  |  |  |  |  |  |
| M02 | **1** | 7 | 7 |  |  |  |  |  |  |  |  |
| M03 |  | **1** | 5 |  |  |  |  |  |  |  |  |
| M04 |  |  | **1** |  |  |  |  |  |  |  |  |
| NºFallos | F1 | F2 | F3 |  |  |  |  |  |  |  |  |
| Tiempo | 2 | 3 | 19 |  |  |  |  |  |  |  |  |

**Tabla resumen de fallos de paginación del Proceso finalizado**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| P02 | | | | | | | | |
| Direcciones | 3 | 8 | 7 | 8 | 6 | 4 | 4 | 9 |
| M01 | 3 | 3 | 3 | 3 | 3 | 4 | 4 | 4 |
| M02 | **1** | 8 | 8 | 8 | 8 | 8 | 8 | 9 |
| M03 |  | 1 | 7 | 7 | 7 | 7 | 7 | 7 |
| M04 |  |  | **1** | **1** | 6 | 6 | 6 | 6 |
| NºFallos | F1 | F2 | F3 | - | F4 | F5 | - | F6 |
| Tiempo | 4 | 5 | 13 | 14 | 15 | 16 | 17 | 18 |

**A Tiempo = 28**

En este punto, el proceso P01 finaliza su ejecución con un total 7 fallos de página y dando por finalizada la ejecución de todos los procesos.

**Banda de memoria:**

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19

**Banda temporal:**

**P01 P02 P03 P04 P03 P02 P01**

3 7 3 8 3 1 1 5 5 8 8 7 8 6 4 4 9 5 8 7 8 6 4 4 9 9 9

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28….

**Tabla resumen de fallos de paginación del Proceso finalizado**

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| P01 | | | | | | | | | | | |
| Direcciones | 3 | 7 | 5 | 8 | 7 | 8 | 6 | 4 | 4 | 9 | 9 |
| M01 | 3 | 3 | 3 | 3 | 3 | 3 | 6 | 6 | 6 | 6 | 6 |
| M02 | **1** | 7 | 7 | 7 | 7 | 7 | 7 | 4 | 4 | 4 | 4 |
| M03 |  | **1** | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 9 | 9 |
| M04 |  |  | **1** | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 |
| NºFallos | F1 | F2 | F3 | F4 | - | - | F5 | F6 | - | F7 | - |
| Tiempo | 2 | 3 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 |

**FINALIZA EL PROGRAMA**

Todos los procesos existentes han sido ejecutados y finalizados, podemos calcular el tiempo medio de espera y de retorno del programa para así tener una evaluación del rendimiento del algoritmo utilizado, al igual que una pequeña tabla con toda la información mas relevante de cada proceso.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Ref | Tell | Tej | Tesp | Tret | Ini | Fin | Fallos |
| P01 | 2 | 11 | 15 | 26 | 19 | 28 | 7 |
| P02 | 4 | 8 | 7 | 15 | 13 | 19 | 6 |
| P03 | 6 | 5 | 2 | 7 | 10 | 13 | 4 |
| P04 | 8 | 2 | 0 | 2 | 8 | 10 | 2 |

**Tiempo medio de Retorno:** (26 + 15 + 7 + 2) / 4 = **12.5**

**Tiempo medio de Espera:** (15 + 7 + 2 + 0) / 4 = **6**

# Medicaciones Realizadas

Implementación del Algoritmo SRPT creando funciones especificas para poder llevarlo a cabo.

Modificaciones en el código de reubicación, para que funcione correctamente.

Modificación de los tiempos medios de Ejecución y de Retraso para que muestren los valore como 0.00 en vez de que sean “–“

Modificación en el ancho de las líneas de memoria, tanto grande como pequeña, la de tiempo y en el resumen de fin de los fallos de páginas(Ej\_pantallas)

Mostrar por pantalla el apuntador de la siguiente página a utilizar en los fallos de paginación.