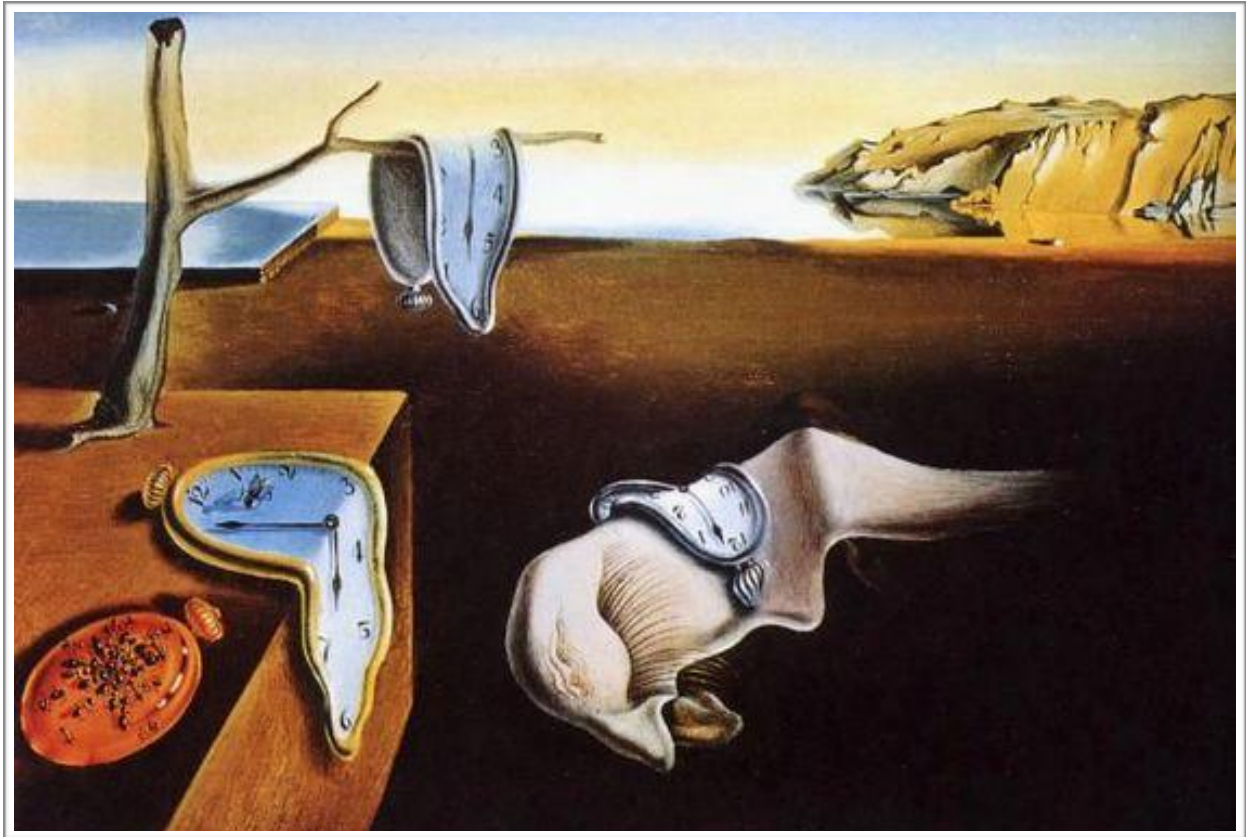


Administración de Memoria

Ni los mejores recuerdos pueden ser accesados al mismo tiempo.



Victor Manuel Cruz Reyes A01152380

Gonzalo Garcia A01281414

Lugo Santillan A01099769

Fernando Gonzales A00568864

ITESM 2017

Información del Programa

El código se toma acompañado.

Información del General

Lenguaje Utilizado: C++ (Version 11, C++11)

Número Total de Líneas: 660

Sin Comentarios: 462

Prologo

La primera vez que vi el trabajo final pensé que era realmente complicado. 10 paginas de especificaciones, objetivos, procesos, requerimientos y entregables se leía un tanto confuso y por lo tanto, difícil. Pensaba, antes de leer el documento, que se trataba de un sistema administrador de memoria el cual es fácil de programar si tomamos únicamente como referencia los algoritmos de remplazo. ¡Hasta hubiéramos logrado hacer el algoritmo Optimo en unas cuantas lineas de Código! Todo esto es posible gracias a nuestra herramienta infalible, las Filas. Aun así, el documento y el trabajo se llegaron a ver imposibles.

Después de pensar confusiones, pues aun no comprendía bien los administradores de memoria, y después de haber escuchado que un grupo termino el trabajo con el uso de 3 clases, volví a recordar los algoritmos de ordenamiento y las filas. De pronto, me di cuenta que la mejor forma de empezar es dividiendo el trabajo en pequeños problemas. Primero creamos la memoria principal, después la llenamos, cuando este llena vienen los algoritmos de remplazo y por ultimo, creamos la memoria secundaria.

Siempre he sabido que la mejor forma de trabajar es diseñando y al ultimo codificar pero debido a la presión del trabajo, tan pronto dividía tareas me ponía a codificarlas. Llegue a tener varios problemas, los cuales encontraba fácilmente pues recurría mucho a técnicas de testing. Pero a pesar de ir por el buen camino, no estaba muy conforme con mi código. Tan pronto avanzaba, encontraba mejores formas de hacer las cosas. Al final comprendí que la mejor forma de avanzar y tener orden es creando clases, pero a pesar de que el código no esta completamente optimizado se que es el correcto. También se que esta muy bien documentado y que lo he probado con algunos casos extremos.

Espero que el código no sea confuso al revisarlo y que este documento provea una mejor comprensión del programa.

Funcionamiento

Empezando con el hecho de que el trabajo se divido en secciones, explicare cada sección y como es que se decidió atacar el problema.

Memoria Principal

Esta era nuestra estructura principal e inicial a crear. No se pensó mucho y al primer momento de leer sobre ella vimos un arreglo. Si un proceso se divide en Paginas y la Memoria Principal se convierte en un Marcos, no era necesario inicializarla en 2048 Bytes sino en 128 Marcos.

Después de leer que era necesario saber el nombre del Proceso y el numero de Pagina, el arreglo se convirtió en una Matriz.

Ingresar Memoria Libre

Teniendo nuestra memoria libre era necesario saber como meter procesos. Realmente es simple, buscas el primer espacio disponible. Al mismo tiempo de saber como hacerlo, nos dimos cuenta que también era necesario dividir el proceso en Paginas e ingresar individualmente. Ademas teníamos que saber cuanto del proceso podría entrar en esta función, por lo que hicimos una variable global que supiera cuantos espacios libres había.

Fila

Si la memoria esta llena, ¿Como sabemos que proceso sacar? Las filas se convirtieron en nuestro algoritmo de remplazo. Si, las filas y no una función especial que buscara al proceso y su pagina. Al principio se pensó que fueran las filas las que supieran que proceso sigue en salir, pero después de algunas pruebas nos dimos cuenta que las filas lo único que necesitan saber es que posición de la memoria principal es la elegida para salir. Distinguir FIFO de LRU no era difícil, pues una simple operación en la fila seria el diferenciador. Si un Proceso es accesado, este obtiene una segunda vida y en la pila significaba mover la posición de esa Pagina hasta el final.

Ingresar Memoria Ocupada

Un simple Fila.Front nos contaba el lugar que debería de ser desalojado. Al saber de la ubicación, la pagina que se encontraba allí era borrada, dejando un espacio libre. Y para evitar mas vueltas, en la misma función se ingresaba la Pagina nueva. Después de probar que la Memoria, junto con los algoritmos de remplazamiento funcionaran, nos dimos la tarea de crear la Memoria Secundaria para que los reemplazados no sean simplemente borrados.

Ingresar Memoria Secundaria (Swapping In)

La Memoria Secundaria es similar a la Principal, tan solo es mas grande.

La principal función, de esta función, es funcionar como transportador entre memorias. Lo primero que hace, sabiendo que posición sacar gracias a la Fila, es copiar los datos de la Memoria Principal al primer espacio libre que encuentre en la Memoria Secundaria. Por ultimo, sale para que la función de Ingresar Memoria Ocupada pueda meter la Pagina que requería espacio.

Tocar Proceso

Ahora que podemos ingresar todos los Procesos, era tiempo de poder accederlos. Es importante saber que un Proceso solo puede ser accedido si este se encuentra en Memoria Principal. Es por ello que esta es la primera sección en la que se busca. Si se encuentra, tan solo se modifica la Fila si se esta utilizando la técnica LRU. Si no esta, se procede a buscarlo en Memoria Secundaria sin antes generar un Page Fault.

Sacar de Memoria Secundaria (Swapping Out)

Se busca la Pagina del Proceso que queremos acceder y al encontrarla, se libera ese espacio de Memoria Secundaria. Después, se busca un espacio libre en Memoria Principal. Si existe Memoria libre esta se usa y si no, se debería de sacar una Pagina de acuerdo al Algoritmo de Ordenamiento. Ya después, en Memoria Principal, esta se accede.

En este Programa, para crear un orden de funciones y no generar confusión, se decidió que esta Función incluyera las mismas instrucciones que Ingresar Memoria Libre y Ocupada. Aquí es donde las clases pueden simplificar el código, pues en este caso no se tendrían que repetir instrucciones similares.

Sacar Proceso

Sin duda alguna, esta es la función mas sencilla de todas. Simplemente recurre toda la estructura y elimina todo rastro (Pagina) del proceso a liberar. La Memoria Principal, Memoria Secundaria y la Fila son los lugares en donde busca.

Matriz de Proceso

Es necesario un seguimiento de los Procesos que entran, pues información debe de ser registrada y actualizada. Esta matriz guarda: Nombre, Tamaño, Tiempo de Entrada, Tiempo de Salida y numero de Page Faults. Al entrar, se registran los primeros 3. Al salir, el cuarto y si llegara a ocurrir un Page Fault (Mientras este corriendo) se actualiza la quinta posición. Cabe destacar que los registros pasan por un sistema de verificación que salvaguarda la integridad y confiabilidad de los datos que se guardan aquí.

Estadísticas

Llegamos a esta función cantando, celebrando y sobretodo saboreando 10 exquisitos puntos que saben como aquella cerveza comprada a la ultima hora permitida por Ley. O tal vez, se sienten como un alivio al poder sumarlos al acumulado final de la materia. Toda esta gloria se debe a que esta función son simples operaciones matemáticas dentro de nuestra Matriz de Procesos.

Otras variables, funciones y operaciones se utilizaron para robustecer el programa. Estas son:

Variables Globales

Algoritmo = Representa el tipo de Algoritmo de remplazo que se esta usando.

Tiempo Global = Marca el tiempo mediante las siguientes reglas (Cargar Pagina, Swapp In/Out = +1 | Accesar/Modificar, Liberar = .1) *Nota: Cuando se hace un Swapp Out no se toma en cuenta el segundo de “Cargar Pagina” pues esta operación la hace.

Swapping global = Cada vez que se usa cualquiera de las dos operaciones.

Imprimir Estructuras

La mejor forma de generar testing es sabiendo que se esta haciendo entre cortinas. Es por eso que se creo una función de impresión para cada estructura.

Reiniciar Estructuras

Cada vez que el programa termina, este se debe de reiniciar para que ahora se utilice un algoritmo diferente. Esta función permite borrar toda la información que se tenia en todas las estructuras usadas (Memoria Principal, Memoria Secundaria, Matriz Procesos y Fila) ademas de reiniciar las variables globales (Marcos Libres, Tiempo Global, Cantidad de Procesos, Swapping Global)

Verificación de Procesos

Son diferentes las situaciones que se deben de tomar en cuenta para poder tener un correcto funcionamiento . Cabe destacar que este programa no verifica el texto del archivo y se tiene por entendido que los comandos vienen bien formateados. Al fin y acabo es el propio sistema operativo el que los genera.

La verificación existe como 2 funciones. La primera busca que al ingresar un nuevo Proceso, este no haya existido antes y el tamaño haya sido el adecuado. Si cumple con las condiciones, el proceso es registrado en la Matriz de Procesos. La segunda, usada principalmente cuando se trata de accesar un Proceso existente, busca que este exista, que no haya sido terminado y que la pagina se busca corresponda a alguna existente.

Ademas de estas funciones especificas, se tiene mucho cuidado al usar las funciones condicionales (If, While, For) siendo muy estrictos en su terminación evitando así instrucciones que alteren el resultado del programa.

Outputs

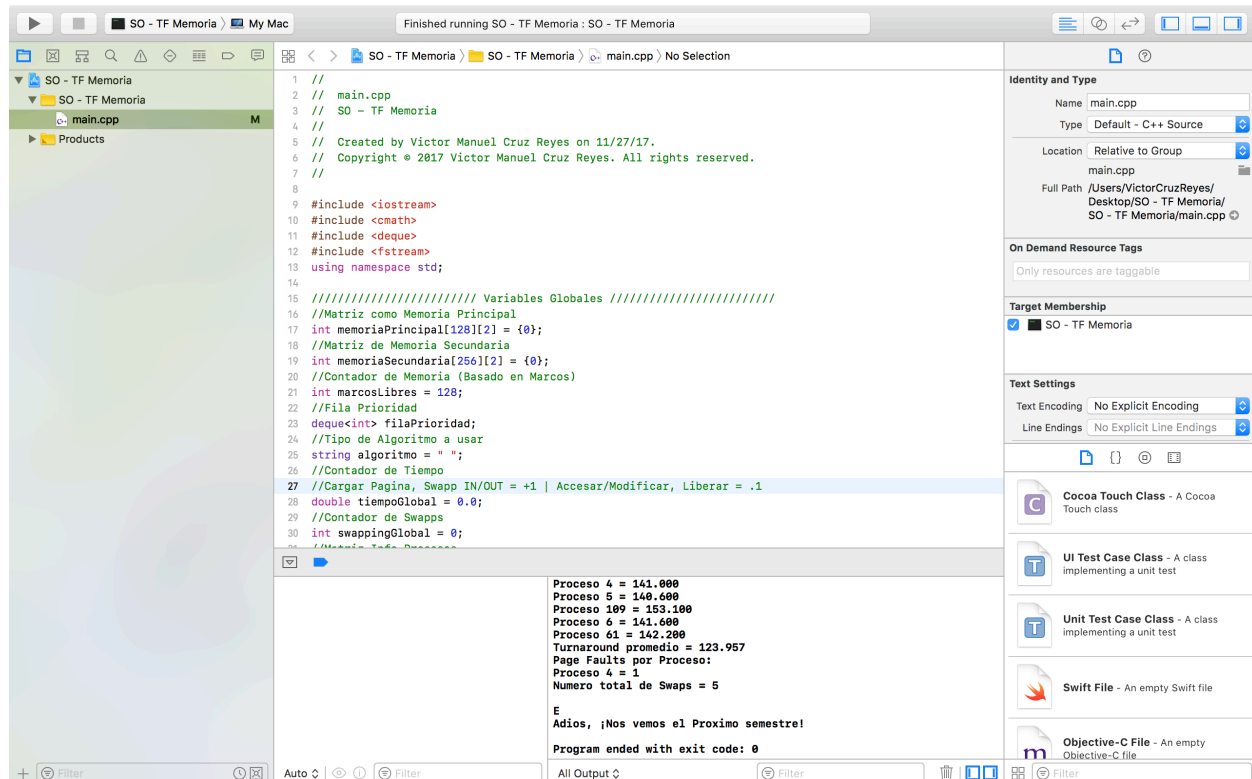
Cuando se ejecuta alguna instrucción importante dentro de la administración de Memoria, como el escribir en Memoria Principal o sacar una Pagina de Memoria Secundaria, es necesario generar los outputs informativos de lo que este pasando. Ademas, cuando ocurre una situación irregular, se informa claramente del error y el porque o como fue causado, como el quedarse sin Memoria Secundaria.

Main

Es un ciclo que permite utilizar los dos tipos de algoritmos con la misma información. Dentro de este, se lee el archivo. Primero se lee el tipo de instrucción y dependiendo de este, mediante un switch, se procede a leer las variables respectivas. Una vez que se leyó todo este archivo, este es reinicado junto con las estructuras para correr con un algoritmo diferente.

Código Corriendo

El código se entrega en documento aparte simplemente para poder apreciar y conservar el diseño de la fuente de Xcode.



Input

P 32 2

P 48 3

P 63 4

L 3

P 80 5

P 1744 109

P 96 6

A 16 2 0

A 63 4 0

A 62 4 1

A 1 109 1

C

P 96 61

L 2

A 8 5 0

A 8 4 0

L 5

L 2

L 4

L 6

L 61

L 109

F

A 2 2 0

P 2049 2049

F

E

Output

////////// FIFO //////////

P 32 2

-Pagina 0 en Marco 0

-Pagina 1 en Marco 1

Se ingresaron todas las Paginas en Memoria Principal

P 48 3

-Pagina 0 en Marco 2

-Pagina 1 en Marco 3

-Pagina 2 en Marco 4

Se ingresaron todas las Paginas en Memoria Principal

P 63 4

-Pagina 0 en Marco 5

-Pagina 1 en Marco 6

-Pagina 2 en Marco 7

-Pagina 3 en Marco 8

Se ingresaron todas las Paginas en Memoria Principal

L 3

Memoria Principal:

Marco 2 liberado

Marco 3 liberado

Marco 4 liberado

Memoria Secundaria:

P 80 5

-Pagina 0 en Marco 2

-Pagina 1 en Marco 3

-Pagina 2 en Marco 4

-Pagina 3 en Marco 9

-Pagina 4 en Marco 10

Se ingresaron todas las Paginas en Memoria Principal

P 1744 109

-Pagina 0 en Marco 11

-Pagina 1 en Marco 12

-Pagina 2 en Marco 13

-Pagina 3 en Marco 14

-Pagina 4 en Marco 15

-Pagina 5 en Marco 16

-Pagina 6 en Marco 17

-Pagina 7 en Marco 18

-Pagina 8 en Marco 19

-Pagina 9 en Marco 20

-Pagina 10 en Marco 21

-Pagina 11 en Marco 22

-Pagina 12 en Marco 23

-Pagina 13 en Marco 24

-Pagina 14 en Marco 25

-Pagina 15 en Marco 26

-Pagina 16 en Marco 27

-Pagina 17 en Marco 28

-Pagina 18 en Marco 29

-Pagina 19 en Marco 30

-Pagina 20 en Marco 31

-Pagina 21 en Marco 32

-Pagina 22 en Marco 33

-Pagina 23 en Marco 34

-Pagina 24 en Marco 35

-Pagina 25 en Marco 36

-Pagina 26 en Marco 37
-Pagina 27 en Marco 38
-Pagina 28 en Marco 39
-Pagina 29 en Marco 40
-Pagina 30 en Marco 41
-Pagina 31 en Marco 42
-Pagina 32 en Marco 43
-Pagina 33 en Marco 44
-Pagina 34 en Marco 45
-Pagina 35 en Marco 46
-Pagina 36 en Marco 47
-Pagina 37 en Marco 48
-Pagina 38 en Marco 49
-Pagina 39 en Marco 50
-Pagina 40 en Marco 51
-Pagina 41 en Marco 52
-Pagina 42 en Marco 53
-Pagina 43 en Marco 54
-Pagina 44 en Marco 55
-Pagina 45 en Marco 56
-Pagina 46 en Marco 57
-Pagina 47 en Marco 58
-Pagina 48 en Marco 59
-Pagina 49 en Marco 60
-Pagina 50 en Marco 61
-Pagina 51 en Marco 62
-Pagina 52 en Marco 63
-Pagina 53 en Marco 64
-Pagina 54 en Marco 65
-Pagina 55 en Marco 66
-Pagina 56 en Marco 67

-Pagina 57 en Marco 68
-Pagina 58 en Marco 69
-Pagina 59 en Marco 70
-Pagina 60 en Marco 71
-Pagina 61 en Marco 72
-Pagina 62 en Marco 73
-Pagina 63 en Marco 74
-Pagina 64 en Marco 75
-Pagina 65 en Marco 76
-Pagina 66 en Marco 77
-Pagina 67 en Marco 78
-Pagina 68 en Marco 79
-Pagina 69 en Marco 80
-Pagina 70 en Marco 81
-Pagina 71 en Marco 82
-Pagina 72 en Marco 83
-Pagina 73 en Marco 84
-Pagina 74 en Marco 85
-Pagina 75 en Marco 86
-Pagina 76 en Marco 87
-Pagina 77 en Marco 88
-Pagina 78 en Marco 89
-Pagina 79 en Marco 90
-Pagina 80 en Marco 91
-Pagina 81 en Marco 92
-Pagina 82 en Marco 93
-Pagina 83 en Marco 94
-Pagina 84 en Marco 95
-Pagina 85 en Marco 96
-Pagina 86 en Marco 97
-Pagina 87 en Marco 98

- Pagina 88 en Marco 99
- Pagina 89 en Marco 100
- Pagina 90 en Marco 101
- Pagina 91 en Marco 102
- Pagina 92 en Marco 103
- Pagina 93 en Marco 104
- Pagina 94 en Marco 105
- Pagina 95 en Marco 106
- Pagina 96 en Marco 107
- Pagina 97 en Marco 108
- Pagina 98 en Marco 109
- Pagina 99 en Marco 110
- Pagina 100 en Marco 111
- Pagina 101 en Marco 112
- Pagina 102 en Marco 113
- Pagina 103 en Marco 114
- Pagina 104 en Marco 115
- Pagina 105 en Marco 116
- Pagina 106 en Marco 117
- Pagina 107 en Marco 118
- Pagina 108 en Marco 119

Se ingresaron todas las Paginas en Memoria Principal

P 96 6

- Pagina 0 en Marco 120
- Pagina 1 en Marco 121
- Pagina 2 en Marco 122
- Pagina 3 en Marco 123
- Pagina 4 en Marco 124
- Pagina 5 en Marco 125

Se ingresaron todas las Paginas en Memoria Principal

A 16 2 0

Direccion Virtual: 16 Direccion Real: 16

A 63 4 0

Direccion Virtual: 63 Direccion Real: 143

A 62 4 1

Direccion Virtual: 62 Direccion Real: 142

A 1 109 1

Direccion Virtual: 1 Direccion Real: 177

C

P 96 61

-Pagina 0 en Marco 126

-Pagina 1 en Marco 127

No quedan espacios libres en Memoria Principal

Se procede a Tecnica de Remplazo

Sale pagina 0 del Proceso 2

Posicion en Memoria Secundaria: 0 (Swapp In)

-Pagina 2 en Marco 0

Sale pagina 1 del Proceso 2

Posicion en Memoria Secundaria: 1 (Swapp In)

-Pagina 3 en Marco 1

Sale pagina 0 del Proceso 4

Posicion en Memoria Secundaria: 2 (Swapp In)

-Pagina 4 en Marco 5

Sale pagina 1 del Proceso 4

Posicion en Memoria Secundaria: 3 (Swapp In)

-Pagina 5 en Marco 6

Se ingresaron todas las Paginas en Memoria Principal

L 2

Memoria Principal:

Memoria Secundaria:

Marco 0 liberado

Marco 1 liberado

A 8 5 0

Direccion Virtual: 8 Direccion Real: 40

A 8 4 0

Page Fault: Pagina no esta en Memoria Principal.

Se procede a buscarla en Memoria Secundaria.

-Pagina 0 en Posicion 2 de Memoria Secundaria

7

Sale pagina 2 del Proceso 4

Posicion en Memoria Secundaria: 0 (Swapp In)

-Pagina 0 en Marco 7

Direccion Virtual: 8 Direccion Real: 120

L 5

Memoria Principal:

Marco 2 liberado

Marco 3 liberado

Marco 4 liberado

Marco 9 liberado

Marco 10 liberado

Memoria Secundaria:

L 2

El proceso ya habia sido liberado.

Se rechaza.

L 4

Memoria Principal:

Marco 7 liberado

Marco 8 liberado

Memoria Secundaria:

Marco 0 liberado

Marco 3 liberado

L 6

Memoria Principal:

Marco 120 liberado

Marco 121 liberado

Marco 122 liberado

Marco 123 liberado

Marco 124 liberado

Marco 125 liberado

Memoria Secundaria:

L 61

Memoria Principal:

Marco 0 liberado

Marco 1 liberado

Marco 5 liberado

Marco 6 liberado

Marco 126 liberado

Marco 127 liberado

Memoria Secundaria:

L 109

Memoria Principal:

Marco 11 liberado

Marco 12 liberado

Marco 13 liberado

Marco 14 liberado

Marco 15 liberado

Marco 16 liberado

Marco 17 liberado

Marco 18 liberado

Marco 19 liberado

Marco 20 liberado

Marco 21 liberado

Marco 22 liberado

Marco 23 liberado

Marco 24 liberado

Marco 25 liberado

Marco 26 liberado

Marco 27 liberado

Marco 28 liberado

Marco 29 liberado

Marco 30 liberado

Marco 31 liberado

Marco 32 liberado

Marco 33 liberado

Marco 34 liberado

Marco 35 liberado

Marco 36 liberado

Marco 37 liberado

Marco 38 liberado

Marco 39 liberado
Marco 40 liberado
Marco 41 liberado
Marco 42 liberado
Marco 43 liberado
Marco 44 liberado
Marco 45 liberado
Marco 46 liberado
Marco 47 liberado
Marco 48 liberado
Marco 49 liberado
Marco 50 liberado
Marco 51 liberado
Marco 52 liberado
Marco 53 liberado
Marco 54 liberado
Marco 55 liberado
Marco 56 liberado
Marco 57 liberado
Marco 58 liberado
Marco 59 liberado
Marco 60 liberado
Marco 61 liberado
Marco 62 liberado
Marco 63 liberado
Marco 64 liberado
Marco 65 liberado
Marco 66 liberado
Marco 67 liberado
Marco 68 liberado
Marco 69 liberado

Marco 70 liberado
Marco 71 liberado
Marco 72 liberado
Marco 73 liberado
Marco 74 liberado
Marco 75 liberado
Marco 76 liberado
Marco 77 liberado
Marco 78 liberado
Marco 79 liberado
Marco 80 liberado
Marco 81 liberado
Marco 82 liberado
Marco 83 liberado
Marco 84 liberado
Marco 85 liberado
Marco 86 liberado
Marco 87 liberado
Marco 88 liberado
Marco 89 liberado
Marco 90 liberado
Marco 91 liberado
Marco 92 liberado
Marco 93 liberado
Marco 94 liberado
Marco 95 liberado
Marco 96 liberado
Marco 97 liberado
Marco 98 liberado
Marco 99 liberado
Marco 100 liberado

Marco 101 liberado

Marco 102 liberado

Marco 103 liberado

Marco 104 liberado

Marco 105 liberado

Marco 106 liberado

Marco 107 liberado

Marco 108 liberado

Marco 109 liberado

Marco 110 liberado

Marco 111 liberado

Marco 112 liberado

Marco 113 liberado

Marco 114 liberado

Marco 115 liberado

Marco 116 liberado

Marco 117 liberado

Marco 118 liberado

Marco 119 liberado

Memoria Secundaria:

F

TurnAround por Proceso:

Proceso 2 = 139.900

Proceso 3 = 9.300

Proceso 4 = 143.000

Proceso 5 = 142.600

Proceso 109 = 155.100

Proceso 6 = 143.600

Proceso 61 = 144.200

Turnaround promedio = 125.386

Page Faults por Proceso:

Proceso 4 = 1

Numero total de Swaps = 6

A 2 2 0

El proceso ya habia sido liberado.

Se rechaza.

P 2049 2049

El proceso es mayor al debido (2048 Bytes).

Se rechaza.

F

TurnAround por Proceso:

Proceso 2 = 139.900

Proceso 3 = 9.300

Proceso 4 = 143.000

Proceso 5 = 142.600

Proceso 109 = 155.100

Proceso 6 = 143.600

Proceso 61 = 144.200

Turnaround promedio = 125.386

Page Faults por Proceso:

Proceso 4 = 1

Numero total de Swaps = 6

E

Adios, ¡Nos vemos el Proximo semestre!

////////// LRU //////////

P 32 2

-Pagina 0 en Marco 0

-Pagina 1 en Marco 1

Se ingresaron todas las Paginas en Memoria Principal

P 48 3

-Pagina 0 en Marco 2

-Pagina 1 en Marco 3

-Pagina 2 en Marco 4

Se ingresaron todas las Paginas en Memoria Principal

P 63 4

-Pagina 0 en Marco 5

-Pagina 1 en Marco 6

-Pagina 2 en Marco 7

-Pagina 3 en Marco 8

Se ingresaron todas las Paginas en Memoria Principal

L 3

Memoria Principal:

Marco 2 liberado

Marco 3 liberado

Marco 4 liberado

Memoria Secundaria:

P 80 5

-Pagina 0 en Marco 2

-Pagina 1 en Marco 3

-Pagina 2 en Marco 4

-Pagina 3 en Marco 9

-Pagina 4 en Marco 10

Se ingresaron todas las Paginas en Memoria Principal

P 1744 109

-Pagina 0 en Marco 11

-Pagina 1 en Marco 12

-Pagina 2 en Marco 13

-Pagina 3 en Marco 14

-Pagina 4 en Marco 15

-Pagina 5 en Marco 16

-Pagina 6 en Marco 17

-Pagina 7 en Marco 18

-Pagina 8 en Marco 19

-Pagina 9 en Marco 20

-Pagina 10 en Marco 21

-Pagina 11 en Marco 22

-Pagina 12 en Marco 23

-Pagina 13 en Marco 24

-Pagina 14 en Marco 25

-Pagina 15 en Marco 26

-Pagina 16 en Marco 27

-Pagina 17 en Marco 28

-Pagina 18 en Marco 29

-Pagina 19 en Marco 30

-Pagina 20 en Marco 31

-Pagina 21 en Marco 32

-Pagina 22 en Marco 33

-Pagina 23 en Marco 34

-Pagina 24 en Marco 35

-Pagina 25 en Marco 36

-Pagina 26 en Marco 37

-Pagina 27 en Marco 38
-Pagina 28 en Marco 39
-Pagina 29 en Marco 40
-Pagina 30 en Marco 41
-Pagina 31 en Marco 42
-Pagina 32 en Marco 43
-Pagina 33 en Marco 44
-Pagina 34 en Marco 45
-Pagina 35 en Marco 46
-Pagina 36 en Marco 47
-Pagina 37 en Marco 48
-Pagina 38 en Marco 49
-Pagina 39 en Marco 50
-Pagina 40 en Marco 51
-Pagina 41 en Marco 52
-Pagina 42 en Marco 53
-Pagina 43 en Marco 54
-Pagina 44 en Marco 55
-Pagina 45 en Marco 56
-Pagina 46 en Marco 57
-Pagina 47 en Marco 58
-Pagina 48 en Marco 59
-Pagina 49 en Marco 60
-Pagina 50 en Marco 61
-Pagina 51 en Marco 62
-Pagina 52 en Marco 63
-Pagina 53 en Marco 64
-Pagina 54 en Marco 65
-Pagina 55 en Marco 66
-Pagina 56 en Marco 67
-Pagina 57 en Marco 68

-Pagina 58 en Marco 69
-Pagina 59 en Marco 70
-Pagina 60 en Marco 71
-Pagina 61 en Marco 72
-Pagina 62 en Marco 73
-Pagina 63 en Marco 74
-Pagina 64 en Marco 75
-Pagina 65 en Marco 76
-Pagina 66 en Marco 77
-Pagina 67 en Marco 78
-Pagina 68 en Marco 79
-Pagina 69 en Marco 80
-Pagina 70 en Marco 81
-Pagina 71 en Marco 82
-Pagina 72 en Marco 83
-Pagina 73 en Marco 84
-Pagina 74 en Marco 85
-Pagina 75 en Marco 86
-Pagina 76 en Marco 87
-Pagina 77 en Marco 88
-Pagina 78 en Marco 89
-Pagina 79 en Marco 90
-Pagina 80 en Marco 91
-Pagina 81 en Marco 92
-Pagina 82 en Marco 93
-Pagina 83 en Marco 94
-Pagina 84 en Marco 95
-Pagina 85 en Marco 96
-Pagina 86 en Marco 97
-Pagina 87 en Marco 98
-Pagina 88 en Marco 99

- Pagina 89 en Marco 100
- Pagina 90 en Marco 101
- Pagina 91 en Marco 102
- Pagina 92 en Marco 103
- Pagina 93 en Marco 104
- Pagina 94 en Marco 105
- Pagina 95 en Marco 106
- Pagina 96 en Marco 107
- Pagina 97 en Marco 108
- Pagina 98 en Marco 109
- Pagina 99 en Marco 110
- Pagina 100 en Marco 111
- Pagina 101 en Marco 112
- Pagina 102 en Marco 113
- Pagina 103 en Marco 114
- Pagina 104 en Marco 115
- Pagina 105 en Marco 116
- Pagina 106 en Marco 117
- Pagina 107 en Marco 118
- Pagina 108 en Marco 119

Se ingresaron todas las Paginas en Memoria Principal

P 96 6

- Pagina 0 en Marco 120
- Pagina 1 en Marco 121
- Pagina 2 en Marco 122
- Pagina 3 en Marco 123
- Pagina 4 en Marco 124
- Pagina 5 en Marco 125

Se ingresaron todas las Paginas en Memoria Principal

A 16 2 0

Direccion Virtual: 16 Direccion Real: 16

A 63 4 0

Direccion Virtual: 63 Direccion Real: 143

A 62 4 1

Direccion Virtual: 62 Direccion Real: 142

A 1 109 1

Direccion Virtual: 1 Direccion Real: 177

C

P 96 61

-Pagina 0 en Marco 126

-Pagina 1 en Marco 127

No quedan espacios libres en Memoria Principal

Se procede a Tecnica de Remplazo

Sale pagina 0 del Proceso 2

Posicion en Memoria Secundaria: 0 (Swapp In)

-Pagina 2 en Marco 0

Sale pagina 0 del Proceso 4

Posicion en Memoria Secundaria: 1 (Swapp In)

-Pagina 3 en Marco 5

Sale pagina 1 del Proceso 4

Posicion en Memoria Secundaria: 2 (Swapp In)

-Pagina 4 en Marco 6

Sale pagina 2 del Proceso 4

Posicion en Memoria Secundaria: 3 (Swapp In)

-Pagina 5 en Marco 7

Se ingresaron todas las Paginas en Memoria Principal

L 2

Memoria Principal:

Marco 1 liberado

Memoria Secundaria:

Marco 0 liberado

A 8 5 0

Direccion Virtual: 8 Direccion Real: 40

A 8 4 0

Page Fault: Pagina no esta en Memoria Principal.

Se procede a buscarla en Memoria Secundaria.

-Pagina 0 en Posicion 1 de Memoria Secundaria

-Pagina 0 en Marco 1

Direccion Virtual: 8 Direccion Real: 24

L 5

Memoria Principal:

Marco 2 liberado

Marco 3 liberado

Marco 4 liberado

Marco 9 liberado

Marco 10 liberado

Memoria Secundaria:

L 2

El proceso ya habia sido liberado.

Se rechaza.

L 4

Memoria Principal:

Marco 1 liberado

Marco 8 liberado

Memoria Secundaria:

Marco 2 liberado

Marco 3 liberado

L 6

Memoria Principal:

Marco 120 liberado

Marco 121 liberado

Marco 122 liberado

Marco 123 liberado

Marco 124 liberado

Marco 125 liberado

Memoria Secundaria:

L 61

Memoria Principal:

Marco 0 liberado

Marco 5 liberado

Marco 6 liberado

Marco 7 liberado

Marco 126 liberado

Marco 127 liberado

Memoria Secundaria:

L 109

Memoria Principal:

Marco 11 liberado

Marco 12 liberado
Marco 13 liberado
Marco 14 liberado
Marco 15 liberado
Marco 16 liberado
Marco 17 liberado
Marco 18 liberado
Marco 19 liberado
Marco 20 liberado
Marco 21 liberado
Marco 22 liberado
Marco 23 liberado
Marco 24 liberado
Marco 25 liberado
Marco 26 liberado
Marco 27 liberado
Marco 28 liberado
Marco 29 liberado
Marco 30 liberado
Marco 31 liberado
Marco 32 liberado
Marco 33 liberado
Marco 34 liberado
Marco 35 liberado
Marco 36 liberado
Marco 37 liberado
Marco 38 liberado
Marco 39 liberado
Marco 40 liberado
Marco 41 liberado
Marco 42 liberado

Marco 43 liberado
Marco 44 liberado
Marco 45 liberado
Marco 46 liberado
Marco 47 liberado
Marco 48 liberado
Marco 49 liberado
Marco 50 liberado
Marco 51 liberado
Marco 52 liberado
Marco 53 liberado
Marco 54 liberado
Marco 55 liberado
Marco 56 liberado
Marco 57 liberado
Marco 58 liberado
Marco 59 liberado
Marco 60 liberado
Marco 61 liberado
Marco 62 liberado
Marco 63 liberado
Marco 64 liberado
Marco 65 liberado
Marco 66 liberado
Marco 67 liberado
Marco 68 liberado
Marco 69 liberado
Marco 70 liberado
Marco 71 liberado
Marco 72 liberado
Marco 73 liberado

Marco 74 liberado
Marco 75 liberado
Marco 76 liberado
Marco 77 liberado
Marco 78 liberado
Marco 79 liberado
Marco 80 liberado
Marco 81 liberado
Marco 82 liberado
Marco 83 liberado
Marco 84 liberado
Marco 85 liberado
Marco 86 liberado
Marco 87 liberado
Marco 88 liberado
Marco 89 liberado
Marco 90 liberado
Marco 91 liberado
Marco 92 liberado
Marco 93 liberado
Marco 94 liberado
Marco 95 liberado
Marco 96 liberado
Marco 97 liberado
Marco 98 liberado
Marco 99 liberado
Marco 100 liberado
Marco 101 liberado
Marco 102 liberado
Marco 103 liberado
Marco 104 liberado

Marco 105 liberado

Marco 106 liberado

Marco 107 liberado

Marco 108 liberado

Marco 109 liberado

Marco 110 liberado

Marco 111 liberado

Marco 112 liberado

Marco 113 liberado

Marco 114 liberado

Marco 115 liberado

Marco 116 liberado

Marco 117 liberado

Marco 118 liberado

Marco 119 liberado

Memoria Secundaria:

F

TurnAround por Proceso:

Proceso 2 = 139.900

Proceso 3 = 9.300

Proceso 4 = 141.000

Proceso 5 = 140.600

Proceso 109 = 153.100

Proceso 6 = 141.600

Proceso 61 = 142.200

Turnaround promedio = 123.957

Page Faults por Proceso:

Proceso 4 = 1

Numero total de Swaps = 5

A 2 2 0

El proceso ya habia sido liberado.

Se rechaza.

P 2049 2049

El proceso es mayor al debido (2048 Bytes).

Se rechaza.

F

TurnAround por Proceso:

Proceso 2 = 139.900

Proceso 3 = 9.300

Proceso 4 = 141.000

Proceso 5 = 140.600

Proceso 109 = 153.100

Proceso 6 = 141.600

Proceso 61 = 142.200

Turnaround promedio = 123.957

Page Faults por Proceso:

Proceso 4 = 1

Numero total de Swaps = 5

E

Adios, ¡Nos vemos el Proximo semestre!

Comparativa

Como comparativa se utiliza el ultimo output de la función F de cada algoritmo.

FIFO	LRU
TurnAround por Proceso:	TurnAround por Proceso:
Proceso 2 = 139.900	Proceso 2 = 139.900
Proceso 3 = 9.300	Proceso 3 = 9.300
Proceso 4 = 143.000	Proceso 4 = 141.000
Proceso 5 = 142.600	Proceso 5 = 140.600
Proceso 109 = 155.100	Proceso 109 = 153.100
Proceso 6 = 143.600	Proceso 6 = 141.600
Proceso 61 = 144.200	Proceso 61 = 142.200
Turnaround promedio = 125.386	Turnaround promedio = 123.957
Page Faults por Proceso:	Page Faults por Proceso:
Proceso 4 = 1	Proceso 4 = 1
Numero total de Swaps = 6	Numero total de Swaps = 5