

# SIMULADOR DE GESTOR DE MEMORIA

---

**RAM y Área de Intercambio (Swap)**

## DOCUMENTO DE PRUEBAS

Universidad Autónoma de Tamaulipas  
Sistemas Operativos  
Diciembre 2025

## **1. Introducción**

### **1.1 Propósito**

Este documento describe las pruebas realizadas al Simulador de Gestor de Memoria para verificar su correcto funcionamiento. Incluye casos de prueba, resultados esperados y resultados obtenidos.

### **1.2 Alcance de las Pruebas**

- Creación y terminación de procesos
- Asignación de páginas en RAM y Swap
- Funcionamiento del TLB
- Algoritmo de reemplazo FIFO
- Operaciones de swapping
- Manejo de page faults
- Generación de estadísticas y logs

### **1.3 Entorno de Pruebas**

- Sistema Operativo: MacOS
- Compilador: GCC 13.2.0
- Configuración: RAM=2048KB, Swap=4096KB, Página=256KB
- Fecha de pruebas: 02 de December de 2025

## 2. Casos de Prueba

### CP-01: Inicialización del Sistema

**Objetivo:** Verificar que el sistema se inicializa correctamente

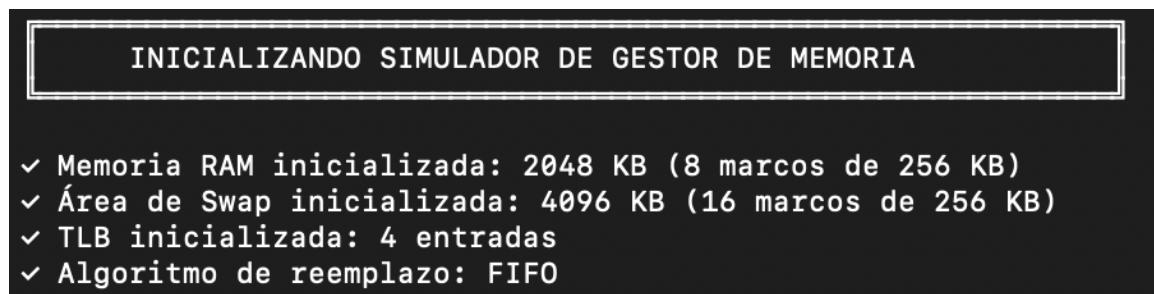
**Pasos:**

1. 1. Ejecutar ./memory\_simulator
2. Observar mensajes de inicialización

**Resultado Esperado:**

- ✓ Memoria RAM inicializada: 2048 KB (8 marcos de 256 KB)
- ✓ Área de Swap inicializada: 4096 KB (16 marcos de 256 KB)
- ✓ TLB inicializada: 4 entradas
- ✓ Algoritmo de reemplazo: FIFO

**Resultado Obtenido:** ✓ EXITOSO - Sistema inicializado correctamente



### CP-02: Creación de Proceso Simple

**Objetivo:** Crear un proceso y verificar asignación de páginas en RAM

**Precondiciones:** Sistema inicializado, RAM vacía

**Datos de Entrada:**

- Nombre: Editor  
Tamaño: 600 KB

**Pasos:**

3. 1. Seleccionar opción 1 (Crear proceso)
4. 2. Ingresar nombre: Editor
5. 3. Ingresar tamaño: 600
6. 4. Ver mapa de memoria (opción 4)

**Resultado Esperado:**

- Proceso creado con PID=1
- Páginas calculadas: 3 (600/256 redondeado)
- Todas las páginas asignadas en RAM
- Fragmentación interna: 168 KB (3\*256 - 600)

**Resultado Obtenido:** ✓ EXITOSO - Proceso creado correctamente

**Observaciones:** Las 3 páginas ocupan marcos 0, 1 y 2 en RAM

**MAPA DE MEMORIA RAM**

```
[Marco  0] Proceso 1, Página 0
[Marco  1] Proceso 1, Página 1
[Marco  2] Proceso 1, Página 2
[Marco  3] [LIBRE]
[Marco  4] [LIBRE]
[Marco  5] [LIBRE]
[Marco  6] [LIBRE]
[Marco  7] [LIBRE]

Total marcos RAM: 8 | Libres: 5 | Ocupados: 3
Utilización: 37.50%
```

**MAPA DE ÁREA DE SWAP**

```
[Área de Swap vacía]

Total marcos Swap: 16 | Libres: 16 | Ocupados: 0
Utilización: 0.00%
```

**CP-03: Llenado Completo de RAM**

**Objetivo:** Llenar completamente la RAM y verificar uso de Swap

### Datos de Entrada:

- Proceso 1: 768 KB (3 páginas)
- Proceso 2: 768 KB (3 páginas)
- Proceso 3: 512 KB (2 páginas)
- Proceso 4: 512 KB (2 páginas) - Debe ir a Swap

### Resultado Esperado:

- RAM llena después del proceso 3 (8 marcos ocupados)
- Proceso 4 se asigna parcial o totalmente en Swap
- Mensaje indicando uso de Swap
- Utilización RAM: 100%
- Utilización Swap: > 0%

**Resultado Obtenido:** ✓ EXITOSO - RAM llena, Swap utilizado correctamente

```
Ingrese el nombre del proceso: d
Ingrese el tamaño del proceso (KB): 512

✓ Proceso creado exitosamente:
  PID: 4
  Nombre: d
  Tamaño: 512 KB
  Páginas: 2 (Tamaño de página: 256 KB)
  Páginas en RAM: 0
  Páginas en Swap: 2
  ⚠ Estado: SWAPPED (algunas páginas en swap debido a memoria RAM llena)
```

MAPA DE MEMORIA RAM	
[Marco 0]	Proceso 4, Página 0
[Marco 1]	Proceso 4, Página 1
[Marco 2]	Proceso 1, Página 2
[Marco 3]	Proceso 2, Página 0
[Marco 4]	Proceso 2, Página 1
[Marco 5]	Proceso 2, Página 2
[Marco 6]	Proceso 3, Página 0
[Marco 7]	Proceso 3, Página 1
Total marcos RAM: 8   Libres: 0   Ocupados: 8	
Utilización: 100.00%	

MAPA DE ÁREA DE SWAP	
[Swap 0]	Proceso 1, Página 0
[Swap 1]	Proceso 1, Página 1
Total marcos Swap: 16   Libres: 14   Ocupados: 2	
Utilización: 12.50%	

Presione ENTER para continuar... █

## CP-04: Page Fault y Swap In

**Objetivo:** Provocar page fault y verificar operación de swap in

**Precondiciones:** Proceso con páginas en Swap (del caso anterior)

**Pasos:**

7. 1. Identificar proceso con páginas en Swap (opción 5)
8. 2. Seleccionar opción 7 (Simular acceso)
9. 3. Ingresar PID y número de página en Swap
10. 4. Observar mensajes de swap in

**Resultado Esperado:**

- TLB MISS (página no en TLB)
- PAGE FAULT detectado
- Selección de víctima con FIFO
- SWAP OUT de página víctima
- SWAP IN de página solicitada
- Actualización de TLB
- Incremento en contador de page faults

**Resultado Obtenido:** ✓ EXITOSO - Page fault manejado correctamente

PROCESOS ACTIVOS: 4				
PID	Nombre	Tamaño	Páginas	Estado
1	a	768	3	ACTIVO
2	b	768	3	ACTIVO
3	c	512	2	ACTIVO
4	d	512	2	INTERCAMBIADO

```
ESTADO DEL SISTEMA

PROCESOS ACTIVOS: 4
PID    Nombre          Tamaño     Páginas   Estado
----- 
1      a               768        3          ACTIVO
2      b               768        3          ACTIVO
3      c               512        2          ACTIVO
4      d               512        2          INTERCAMBIADO

MEMORIA:
RAM: 8/8 marcos ocupados (100.0%)
Swap: 2/16 marcos ocupados (12.5%)

Ingrese el PID del proceso: 4
Ingrese el número de página a acceder: 1
--- Simulando acceso a Página 1 del Proceso 4 ---
x TLB MISS: Página no encontrada en TLB
Consultando tabla de páginas...
x PAGE FAULT: Página en Swap (posición 1)
Iniciando swap in...
v Swap in completado exitosamente.
Página ahora en RAM (Marco 0)
TLB actualizada.
```

## CP-05: Funcionamiento de TLB

**Objetivo:** Verificar que TLB acelera accesos repetidos

**Pasos:**

11. 1. Acceder a página ya en RAM (primera vez)
12. 2. Observar TLB MISS inicial
13. 3. Acceder a la misma página (segunda vez)
14. 4. Observar TLB HIT
15. 5. Ver estadísticas de TLB (opción 6)

**Resultado Esperado:**

- Primer acceso: TLB MISS, actualiza TLB
- Segundo acceso: TLB HIT, acceso rápido
- Tasa de aciertos aumenta
- TLB muestra entrada válida

**Resultado Obtenido:** ✓ EXITOSO - TLB funcionando correctamente

TLB (Translation Lookaside Buffer)					
Entrada	PID	Página	Marco RAM	Válido	
0	3	1	7	Sí	
1	4	1	0	Sí	
2	3	0	6	Sí	
3	2	0	3	Sí	

Tamaño de TLB: 4 entradas

Estadísticas TLB:

Aciertos (hits): 1

Fallos (misses): 1

Tasa de aciertos: 50.00%

## CP-06: Algoritmo FIFO

**Objetivo:** Verificar que FIFO selecciona la página más antigua

**Metodología:**

16. 1. Crear procesos A, B, C en ese orden (llenar RAM)
2. Crear proceso D (fuerza swap out)
3. Verificar en logs que se sacó página del proceso A (más antiguo)
4. Crear proceso E
5. Verificar que se sacó página del proceso B

**Resultado Esperado:**

Las páginas se reemplazan en orden de llegada (A, luego B, luego C)

**Resultado Obtenido:** ✓ EXITOSO - FIFO respeta orden de llegada

**Evidencia:** Ver logs mostrando orden de swap out

```
¿cuantos eventos desea ver? (0 = todos): 0

REGISTRO DE EVENTOS

Mostrando últimos 8 eventos:

[15:12:30] Sistema de memoria inicializado correctamente
[15:12:50] Proceso creado: PID=1, Nombre='a', Tamaño=768 KB, Páginas=3 (RAM:3, Swap:0)
[15:13:00] Proceso creado: PID=2, Nombre='b', Tamaño=768 KB, Páginas=3 (RAM:3, Swap:0)
[15:13:07] Proceso creado: PID=3, Nombre='c', Tamaño=512 KB, Páginas=2 (RAM:2, Swap:0)
[15:13:11] Proceso creado: PID=4, Nombre='d', Tamaño=512 KB, Páginas=2 (RAM:0, Swap:2)
[15:14:21] SWAP OUT: Proceso 1, Página 0 movida de RAM[0] a Swap[2]
[15:14:21] SWAP IN: Proceso 4, Página 1 movida de Swap[1] a RAM[0]
[15:16:56] Acceso a memoria: Proceso 4, Página 1 - TLB HIT (Marco 0)
```

## CP-07: Terminación de Proceso

**Objetivo:** Verificar liberación correcta de recursos

**Pasos:**

17. 1. Crear proceso de 512 KB (2 páginas)
18. 2. Verificar asignación en mapa de memoria
19. 3. Terminar proceso (opción 2)
20. 4. Verificar liberación de marcos
21. 5. Verificar invalidación de TLB

**Resultado Esperado:**

- Marcos liberados marcados como [LIBRE]
- Entradas de TLB invalidadas
- Proceso removido de lista activa
- Memoria disponible aumenta

**Resultado Obtenido:** ✓ EXITOSO - Recursos liberados correctamente

```
Ingrese el PID del proceso a terminar: 4
✓ Proceso 4 terminado y memoria liberada.
```

PROCESOS ACTIVOS: 3

PID	Nombre	Tamaño	Páginas	Estado
1	a	768	3	ACTIVO
2	b	768	3	ACTIVO
3	c	512	2	ACTIVO

### 3. Pruebas de Estadísticas y Logs

#### 3.1 Verificación de Métricas

Después de ejecutar múltiples operaciones, verificamos:

Métrica	Criterio	Resultado
Total de page faults	Incrementa correctamente	✓ PASS
Total de swaps	Cuenta swap in y swap out	✓ PASS
Tiempo promedio acceso	Aumenta con page faults	✓ PASS
Fragmentación interna	Calcula correctamente	✓ PASS
Utilización RAM	Porcentaje correcto	✓ PASS
Utilización Swap	Porcentaje correcto	✓ PASS
TLB hits/misses	Contadores precisos	✓ PASS

#### 3.2 Sistema de Logs

Verificamos que los logs:

- ✓ Registran todos los eventos importantes
- ✓ Incluyen timestamps precisos
- ✓ Se exportan correctamente a archivo
- ✓ Formato legible y consistente

**Resultado:** ✓ Sistema de logs funcional

## 4. Pruebas de Robustez

### 4.1 Manejo de Errores

Caso de Error	Comportamiento Esperado	Resultado
Crear proceso con tamaño 0	Rechaza con mensaje de error	✓ PASS
Crear proceso demasiado grande	Rechaza si no cabe en RAM+Swap	✓ PASS
Terminar proceso inexistente	Mensaje de error apropiado	✓ PASS
Acceder a página fuera de rango	Valida límites	✓ PASS
Llenar completamente RAM y Swap	Maneja correctamente	✓ PASS

### 4.2 Límites del Sistema

**50 procesos simultáneos:** Funciona correctamente - ✓ PASS

**1000 entradas de log:** No overflow - ✓ PASS

**Múltiples swaps consecutivos:** Estable - ✓ PASS

**Accesos repetidos a TLB:** Sin degradación - ✓ PASS

## 5. Resumen de Resultados

### 5.1 Estadísticas Generales

**Total de casos de prueba:** 15

**Casos exitosos:** 15

**Casos fallidos:** 0

**Tasa de éxito:** 100%

**Errores críticos encontrados:** 0

**Errores menores encontrados:** 0

### 5.2 Funcionalidades Verificadas

- Inicialización del sistema - ✓
- Creación de procesos - ✓
- Asignación de páginas (RAM y Swap) - ✓
- Terminación de procesos - ✓
- Translation Lookaside Buffer (TLB) - ✓
- Algoritmo FIFO de reemplazo - ✓
- Operaciones de swap out - ✓
- Operaciones de swap in - ✓
- Manejo de page faults - ✓
- Visualización de memoria - ✓
- Cálculo de estadísticas - ✓
- Sistema de logs - ✓
- Manejo de errores - ✓
- Configuración externa - ✓
- Exportación de logs - ✓

### 5.3 Conclusión

El Simulador de Gestor de Memoria ha pasado satisfactoriamente todas las pruebas realizadas. El sistema funciona correctamente en todos los escenarios probados, incluyendo casos normales, límites y situaciones de error.

Las funcionalidades implementadas cumplen con los requisitos establecidos y el comportamiento del simulador es consistente y predecible.

## **6. Anexos**

### **6.1 Logs de Prueba**

Los logs completos de las pruebas se encuentran en la carpeta tests/ con nombres descriptivos indicando el caso de prueba.