

**SISTEMAS OPERATIVOS 2025-2S**  
**EJERCICIO DE CONCEPTOS DE GESTIÓN DE MEMORIA**  
**EJERCICIO PRÁCTICO DEL 22%**

Escriba un programa en C que realice la traducción de direcciones de memoria en un sistema que tiene un espacio virtual de direccionamiento de 32 bits con páginas de 4 KiB. El programa debe leer de manera indefinida y hasta que el usuario pulse la letra «**s**», una dirección en decimal y mostrar: (a) el número de página en decimal y en binario, (b) el desplazamiento dentro de la página en decimal y en binario, (c) el tiempo en segundos que tomó la operación y (d) si la traducción produce un TLB Hit o un TLB Miss. Para simular el TLB **debe** usar memoria en el segmento *heap* del proceso (**obligatorio**). Implemente toda la lógica de la traducción usando el TLB como caché, de acuerdo con los conceptos explicados en clase.

Requisitos obligatorios en la implementación de la solución.

1. TLB debe ser implementado en el segmento *heap* del proceso.
2. Tamaño máximo en Bytes del TLB: 300 Bytes.
3. Máximo de entradas que se pueden guardar en el TLB: 5
4. Una entrada en TLB consta (mínimamente) de la siguiente información: dirección de memoria en decimal, número de página en decimal, desplazamiento en decimal, número de página en binario y desplazamiento en binario.
5. La política de reemplazo en el TLB estará basada en la política LRU.
6. Cuando se reemplace una entrada en TLB, se debe mostrar la dirección base de memoria de la entrada que está siendo reemplazada. Cuando no haya reemplazo, se debe mostrar **0x0** (ver ejemplo de ejecución).
7. No se permite el uso de arreglos (de cualquier dimensión), estructuras (struct), uniones (union) y tipos definidos por el usuario (typedef) **para implementar y gestionar el TLB**. Toda la gestión del TLB debe ser realizada con variables apuntadores (referencias). No se permite la gestión del TLB con índices de arreglos, estructuras, uniones y/o tipos definidos por el usuario.
8. Máximo de variables apuntadores en la gestión del TLB: 3 por función que necesite manipular el TLB.
9. Las marcas de tiempo deben ser obtenidas con la system call *gettimeofday()*.

**Continua en la siguiente página...**

10. Se debe mostrar la dirección de memoria donde inicial el TLB y la dirección de memoria donde termina el TLB.
11. Si una dirección referenciada no es válida en el espacio de direcciones, se debe imprimir el mensaje «**Page Fault**».
12. Debe existir una función que permita convertir un número en decimal a binario. Esta función debe hacer uso del operador de desplazamiento a la derecha `>>` y el operador `&` (AND) *bitwise* de C y la función se debe usar en el proceso de conversión de decimal a binario.
13. Debe existir una función que permita convertir un número en binario a decimal. Esta función debe calcular el número en decimal como la suma acumulada de elevar 2 a la posición del dígito, para todo dígito igual a 1. Esta función se debe usar en el proceso de conversión de binario a decimal (en caso de que se requiera).
14. Debe usar una variable para la representación de la página en binario.
15. Debe usar una variable para la representación del desplazamiento en binario.
16. Liberar/cerrar adecuadamente los recursos usados. Recuerde que en C no hay *garbage collector* (sorry not sorry team Java).
17. La salida de la ejecución del proceso debe ser como se indica en el **EJEMPLO DE EJECUCIÓN**.

**Continua en la siguiente página...**

## ENTREGABLES, SUSTENTACIÓN Y CONDICIONES DE CALIFICACIÓN

1. Se debe entregar **ÚNICAMENTE** un (1) archivo de código escrito en lenguaje C. El archivo de código fuente deben ser **obligatoriamente extensión .c** (no .pdf, no .docx, no .txt, etc.) y deben estar correctamente formateados: bloques de código con sangría correcta y líneas de código máximo de 80 caracteres. Plazo máximo para entrega en plataforma: **lunes 1 de diciembre de 2025 a las 8am.**
2. La nota total del 22% está dada por los siguientes dos componentes.
  - 2.1 Cuestionario en UNVirtual que se responde de manera presencial e individual en la fecha y hora programada de las sesiones prácticas del curso: **lunes 1 de diciembre de 2025 en el horario que le corresponde. Valor 60%.**
  - 2.2 Ejecución y revisión del código (nota de la pareja de estudiantes). **Valor 40%.**
3. **UNA SOLA ENTREGA POR PAREJA DE ESTUDIANTES EN UNVirtual.** No se admiten entregas por fuera del horario ni por correo electrónico. **Debe indicar en los comentarios de la entrega en UNVirtual, el nombre completo de lo(a)s estudiantes y cualquier instrucción adicional de compilación del código fuente.** Estudiante o pareja de estudiantes que no aparezcan con el registro de la entrega de código en UNVirtual tendrá una calificación de cero en el componente de la nota del 40%.
4. Si en la revisión y ejecución del código se encuentran operaciones, instrucciones o lógica que amerite explicación ampliada por parte de lo(a)s estudiantes involucrado(a)s, se solicitará de manera presencial y verbal a lo(a)s estudiantes, la explicación y sustentación ampliada para confirmar o descartar la nota asignada en el componente del 40%.
5. Resultados positivos para el **control de plagios y similitud de códigos**, anulan la nota completa del 22% **a todo(a)s lo(a)s involucrado(a)s.** Si usted deja su código con visibilidad pública (p. ej.: GitHub) y se lo copian, o lo comparte, es su entera responsabilidad y también acarrearán la misma sanción de lo(a)s estudiantes que copien de él.

**Continua en la siguiente página...**

## EJEMPLO DE EJECUCIÓN

```
#./traducir
Ingrese dirección virtual: 19986
TLB desde 0x00401251 hasta 0x0040136E
TLB Miss
Página: 4
Desplazamiento: 3602
Página en binario: 0000000000000000100
Desplazamiento en binario: 111000010010
Politica de reemplazo: 0x0
Tiempo: 0.000049 segundos
```

```
Ingrese dirección virtual: 19986
TLB desde 0x00401251 hasta 0x0040136E
TLB Hit
Página: 4
Desplazamiento: 3602
Página en binario: 0000000000000000100
Desplazamiento en binario: 111000010010
Politica de reemplazo: 0x0
Tiempo: 0.000037 segundos
```

```
Ingrese dirección virtual: 32768
TLB desde 0x00401251 hasta 0x0040136E
TLB Miss
Página: 8
Desplazamiento: 0
Página en binario: 00000000000000001000
Desplazamiento en binario: 000000000000
Politica de reemplazo: 0x0
Tiempo: 0.000068 segundos
```

### <Otras traducciones...>

```
Ingrese dirección virtual: 32768
TLB desde 0x00401251 hasta 0x0040136E
TLB Miss
Página: 8
Desplazamiento: 0
Página en binario: 00000000000000001000
Desplazamiento en binario: 000000000000
Politica de reemplazo: 0x00401251
Tiempo: 0.000053 segundos
```

```
Ingrese dirección virtual: s
Good bye!
```