

Ejercicio 1

Dirección Lógica	Solicitud	Dirección Física
0-430	R	670
0-150	W	Violación de Seguridad
1-15	R	2315
2-130	W	Desplazamiento fuera del rango
4-25	X	Segmento Faltante

Ejercicio 2

- a) Para el programa A se necesitan $\frac{4300bytes}{128bytes} = 33.59 \rightarrow 34$ páginas como mínimo. Para el programa B se necesitan al menos $\frac{3068bytes}{128bytes} = 23.97 \rightarrow 24$ páginas.
- b) Dado que las páginas son de 128KB, en cada programa se desperdicia en promedio $\frac{2^7}{2} = 64KB$ por página por fragmentación interna. El método por paginación no desperdicia memoria por fragmentación externa.
- c) Dado que hay $34 + 24 = 58$ páginas como mínimo, 34 del programa A y 24 del programa B, entonces la tabla de paginación tendrá un tamaño de al menos $58 * 8bytes = 464$ bytes.

Ejercicio 3

La instrucción LDR en arquitectura ARM permite un formato que incluye cargar un valor constante de 32-bits desde la versión ARMv6T2. Esto es posible porque la arquitectura reserva un espacio de memoria al final de cada sección de código llamado `literal pool`. Su propósito es justamente cargar valores constantes o direcciones de etiquetas, que su tamaño está fuera del rango de las instrucciones MOV y MVN, que luego serán cargados a un registro. Cuando se llama a esta instrucción con un valor constante de 32-bits, el ensamblador realiza lo siguiente:

- Verifica si el valor no se encuentra ya en alguna `literal pool`. Si es así, carga la dirección de la constante ya existente.
- En caso contrario, coloca el valor en la siguiente `literal pool` disponible y carga esa nueva dirección.

Sin embargo, si la siguiente `literal pool` esta fuera de rango, el ensamblador genera un mensaje de error.