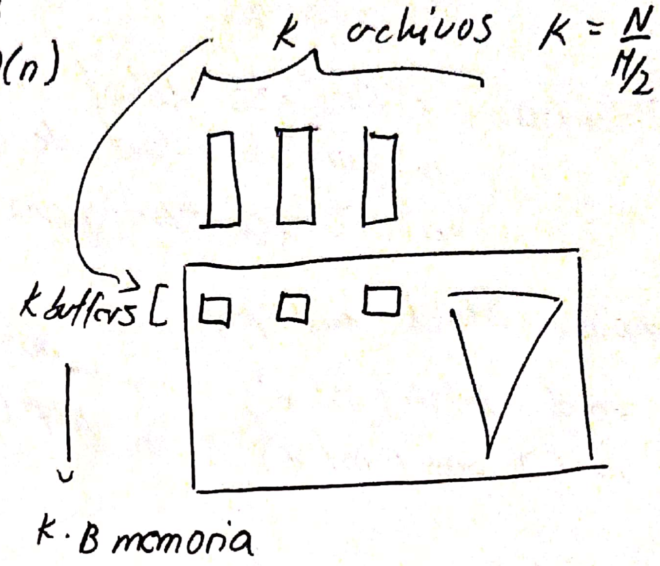
→ costo de este proceso:

El costo por elemento es $\frac{2}{B}$. Si inserto y borro N elementos s.

el costo total es $O(\frac{N}{B}) = O(n)$



$$k \cdot B \leq \frac{M}{2}$$

$$\frac{N}{\frac{M}{2}} B \leq \frac{M}{2}$$

$$\Rightarrow \frac{N}{B} \leq \frac{M^2}{4B^2} \Rightarrow n \leq \frac{m^2}{4}$$

\# Con estas cotas se explica pq aparentemente ordenar es $O(n)$, pero dado el límite de N

↳ $\Theta(n \log_m n) = \Theta(n \log_m m^2) = \Theta(n)$

↳ A costo de soportar solo esas 3 operaciones.

— Análisis Amortizado —

→ Estudiamos el peor caso para una secuencia de k operaciones: $O_1, \dots, O_k$

Def: Costo amortizado (CA) $= \left(\sum_{i=1}^{k} O_i\right) / k$

costo Amortizado ≤ PeorCaso(O)

costo Total = $k \cdot CA$

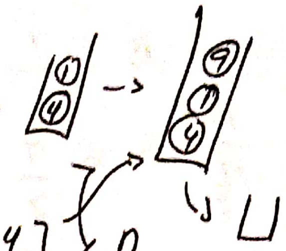
Ej: Peor semana posible:

Lunes	Martes	....	Domingo
Despido			
Divorcio			
⋮			

Obj: Mostrar que el peor caso >> C.A.

Intuición: El peor caso no siempre puede darse en todas las operac.

Técnicas: i) Análisis Global
ii) Contabilidad de Costos
iii) Función de potencial.



-> Ejemplo: Stack multipop.

```
void push(int x);
int[] pop(int k);
```

push(9)
pop(3); -> [9,1,4] } n
⋮

Ahora tomemos n de estas operac.

i) Análisis Global: ¿Cuántos elementos a lo más logra insertar con n operac? n

¿Cuántos elementos uno puede haber sacado en total? $\leq n$

=> A lo más hay $2n = O(n)$

ii) Contabilidad de Costos: (A) Con esto redistribuimos los costos eso n veces nos da un costo de $2n = O(n)$

	Push	Pop(k)
Cuesta:	1	K
(A)	2	0

$\leq 2n$

le cobro entrar y salir.

"Off-Topic"

(iii) Función potencial: (A)'

> ϕ_i = "potencial" acumulado post operac i. $[\phi_i \geq \phi_0]$

> C_i = costo real operac i

secuencia de i operac.

> $\hat{C}_i = C_i + \Delta\phi_i$ # $\Delta\phi_i = \phi_i - \phi_{i-1}$

costo amortizado

En base a lo de antes, el Ejemplo: ϕ_i será el tamaño del stack en el momento i.

> $\phi_i = |S|_i$

(!)

$$\sum_i \hat{C_i} = \sum_i C_i + \sum_i \Delta\phi_i$$ Telescópica :O

$$= \sum_i C_i + \underbrace{\sum_i \phi_i - \phi_{i-1}}$$

$$= \sum_i C_i + \underbrace{(\phi_n - \phi_0)}_{\geq 0 \text{ (A)}} \rightarrow \sum \hat{C_i} \geq \sum C_i$$

Esto quiero saber, si acoto $\sum \hat{C_i}$ entonces tmbn $\sum C_i$:D

→ ¿y en el caso del stack Multipop?

Recordar (!)

$$C_i = \begin{cases} 1 & \text{si push} \\ K & \text{si pop}(K) \end{cases} \qquad \hat{C_i} = \begin{cases} 1 + 1 & \text{pq aumentamos en 1 el stak} \\ K - K & \text{pq borramos } K \text{ cosas del stack} \end{cases}$$

$$\Rightarrow \sum_{i=1}^{n} \hat{C_i} \leq 2n \quad \therefore \sum C_i \leq \sum \hat{C_i} \leq 2n \rightarrow O(n) \text{ :)}$$

\# En mega síntesis, el costo total $\leq 2n$, entonces el costo amortizado $\leq 2 \in O(1)$, esto quiere decir que en total o a la larga (i iteraciones) el costo de las operaciones son constantes.

Remember: Esto no es costo promedio, la idea del amortizado es qué es lo peor que te puede pasar en una secuencia de operaciones.