Estructura de Datos y Algoritmos 2

TP 2 - SECUENCIAS

integrantes
Chanco Castillo Leonid
Ruffini Lucas

Especificación de costos para esta implementación de

`Listas`

- **mapS** `Trabajo` `Profundidad`
- **appendS** `Trabajo` `Profundidad`
- **reduceS** `Trabajo` `Profundidad`
- **scanS** `Trabajo` `Profundidad`

`Arreglos`

- **mapS** `Trabajo` `Profundidad`
- **appendS** `Trabajo` `Profundidad`
- **reduceS** `Trabajo` `Profundidad`
- **scanS** `Trabajo` `Profundidad`

Consideramos:

$f \in O(1)$
n la cantidad de elementos
a_i el elemento i-esimo de la secuencia a:

COSTOS DE SECUENCIAS CON LA IMPLEMENTACION DE LISTAS

MAPS

```
mapL :: (a -> b) -> [a] -> [b]
mapL _ [] = []
mapL f (x:xs) = let (y,ys) = f x ||| mapL f xs
                in (y:ys)
```

`Trabajo`

$$\begin{aligned} W_{mapS}(f,0) &= c_0 \\ W_{mapS}(f,n) &= c_1 + W_{mapS}(f,n-1) + W_f(a_0) \\ &\leq nc_1 + \textstyle\sum_{i=0}^{n-1} W_f(a_i) \qquad\qquad W_f \in O(1) \\ &\leq nc_1 + nc_2 \\ &\leq nc_3 \end{aligned}$$

concluimos

$$W_{mapS} \in O(n)$$

`Profundidad`

$$\begin{aligned} S_{mapS}(f,0) &= c_0 \\ S_{mapS}(f,n) &= \max(S_{mapS}(f,n-1), S_f(a_0)) + c_1 \\ &= \max(\ \max(S_{mapS}(f,n-2), S_f(a_1)) + c_1, S_f(a_0)) + c_1 \\ &\leq nc_1 + \max(S_f(a_1), ..., S_f(a_n)) \qquad\qquad S_f \in O(1) \end{aligned}$$

Podemos concluir

$$S_{mapS} \in O(n)$$

APPENDS

```
appendL :: [a] -> [a] -> [a]
appendL x []     = x
appendL [] y     = y
appendL (x:xs) y = x:(appendL xs y)
```

`Trabajo`

$$\begin{aligned} W_{appendS}(n_1,0) &= c_0 \\ W_{appendS}(0,n_2) &= c_1 \\ W_{appendS}(n_1,n_2) &= c_2 + W_{appendS}(n_1-1) \leq n_1c_1 \end{aligned}$$

Podemos concluir

$$W_{appendS} \in O(n)$$

`Profundidad`

Como la implementacion no esta parelelizada, concluimos

$$S_{appendS}(n) \in O(n)$$

CONTRACTS

```
contractL :: (a -> a -> a) -> [a] -> [a]
contractL f []       = []
contractL f [x]      = [x]
contractL f (x:y:xs) = let (x',xs') = (f x y) ||| (contractL f xs)
                       in (x':xs')
```

`Trabajo`

$$\begin{aligned} W_{contractS}(f,0) &= c_0 \\ W_{contractS}(f,1) &= c_1 \\ W_{contractS}(f,n) &= c_3 + W_{contractS}(f,n-2) + W_f(a_0,a_1) \\ &\leq c_3\tfrac{n}{2} + \textstyle\sum_{i=0}^{\frac{n-2}{2}} W_f(a_{2i},a_{2i+1}) \end{aligned}$$

luego como $W_f \in O(1)$ tenemos que

$$W_{contractS} \in O(n)$$

`Profundidad`

$$\begin{aligned} S_{contractS}(f,0) &= c_0 \\ S_{contractS}(f,1) &= c_1 \\ S_{contractS}(f,n) &= c_3 + \max(S_{contractS}(f,n-2), S_f(a_0,a_1)) \\ &\leq c_3\tfrac{n}{2} + \max(S_f(a_0,a_1), S_f(a_2,a_3), ..., c_4), \quad con: c_4 = \max(c_0,c_1) \end{aligned}$$

Luego como $S_f \in O(1)$ tenemos que

$$S_{contractS} \in O(n)$$

REDUCES

```
reduceL :: (a -> a -> a) -> a -> [a] -> a
reduceL f e [] = e
reduceL f e [x] = f e x
reduceL f e s = let xs = contractL f s
                in reduceL f e xs
```

`Trabajo`

$$\begin{aligned} W_{reduceS}(f,e,0) &= c_0 \\ W_{reduceS}(f,e,1) &= c_1 + W_f(e,a_0) \\ W_{reduceS}(f,e,n) &= c_3 + W_{contractS}(f,n) + W_{reduceS}(f,e,\lceil\tfrac{n}{2}\rceil) \end{aligned}$$

Por suavidad tenemos

$$W_{reduceS}(f,e,n) = W_{reduceS}(f,e,\frac{n}{2}) + W_{contractS}(f,n) + c_3$$

Vamos a aplicar el tercer caso del **Teorema maestro**, sean:

$a = 1, b = 2$ y $W_{contractS}(n) = f(n) \in O(n)$

Calculamos $d = log_b a = log_2 1 = 0$

$W_{contractS}(n) \in \Omega(n^{d+e})$
$W_{contractS}(n) \in \Omega(n^{0+e})$
$con\ e = 1$ se tiene:
$W_{contractS}(n) \in \Omega(n)$

ademas $\exists c < 1, n_0 \in N. \forall n > n_0 : af(n/b) \leq cf(n)$
$f(n/2) \leq cf(n)$
$\frac{n}{2} \leq cn$ Eligiendo $c = \frac{1}{2}$ se cumple

Entonces podemos concluir por el **Teorema Maestro**

$$W_{reduceS}(n) \in O(W_{contractS}(n)) = O(n)$$

`Profundidad`

Como la implementacion no esta parelelizada, concluimos

$$S_{contractS}(n) \in O(n)$$

EXPANDS

```
expandL :: (a -> a -> a) -> [a] -> [a] -> [a]
expandL f [] _ = []
expandL f [x] (y:ys) = [y]
expandL f (x1:x2:xs) (y:ys) = let (x',xs') = f y x1 ||| expandL f xs ys
                              in y:x':xs'
```

`Trabajo`

$$\begin{aligned} W_{expandS}(f,0,n_2) &= c_0 \\ W_{expandS}(f,1,n_2) &= c_1 \\ W_{expandS}(f,n_1,n_2) &= c_2 + W_f(a_0,b_0) + W_{expandS}(f,n_1-2,n_2-1) \\ &\leq n_1c_2 + \textstyle\sum_{i=0}^{\frac{n_1}{2}} W_f(a_{2i},b_i) \end{aligned}$$

Luego como $W_f \in O(1)$ tenemos que

$$S_{expandS} \in O(n)$$

`Profundidad`

$$\begin{aligned} S_{expandS}(f,0,n_2) &= c_0 \\ S_{expandS}(f,1,n_2) &= c_1 \\ S_{expandS}(f,n_1,n_2) &= c_2 + \max(S_f(a_0,b_0), S_{expandS}(f,n_1-2,n_2-1)) \end{aligned}$$

Luego como $S_f \in O(1)$ tenemos que

$$S_{expandS} \in O(n)$$

SCANS

```
scanL :: (a -> a -> a) -> a -> [a] -> ([a],a)
scanL f e []  = ([],e)
scanL f e [x] = ([e],f e x)
scanL f e s   = let xs = contractL f s
                    (s',t) = scanL f e xs
                 in (expandL f s s', t)
```

`Trabajo`

$$\begin{aligned} W_{scanS}(f,e,0) &= c_0 \\ W_{scanS}(f,e,1) &= c_1 + W_f(e,a_0) \\ W_{scanS}(f,e,n) &= c_2 + W_{contractS}(f,n) + W_{scanS}(f,e,\lceil\tfrac{n}{2}\rceil) + W_{expandS}(f,n,\lceil\tfrac{n}{2}\rceil) \\ &\leq n_1 + W_{scanS}(f,e,\lceil\tfrac{n}{2}\rceil) + n_0 \\ &\leq^* W_{scanS}(f,e,\lceil\tfrac{n}{2}\rceil) + n \end{aligned}$$

Al ser una función suave se tiene la recurrencia: $W_{scanS}(\frac{n}{2}) + n$
por el 3er caso del **Teorema Maestro** se concluye

$$W_{scanS}(n) \in O(n)$$

$*W_{contracS}, W_{expandS} \in O(n)$ demostrado en los ejercicios anteriores

`Profundidad`

Como $S_{contractS}(n), S_{exandS}(n) \in O(n)$ y ademas scanS no paraleliza se tiene que:

$$S_{scanS}(n) = W_{scanS}(n) \in O(n)$$

COSTOS DE SECUENCIAS CON LA IMPLEMENTACION DE ARREGLOS

MAPS

```
lengthA :: A.Arr a -> Int
lengthA a = A.length a

mapA :: (a -> b) -> A.Arr a -> A.Arr b
mapA f a = let l = lengthA a
                     in tabulateA (\i -> f(a!i)) l
```

`Trabajo`

Dado que lengthA $\in O(1)$

$$\begin{aligned} W_{mapS}(f,n) &= W_{lengthA}(n) + W_{tabulateA}(f,n) \\ &= c_0 + \textstyle\sum_{i=0}^{n-1} W_f(a_i), \qquad\qquad W_f \in O(1) \\ &= c_0 + c_1 n \end{aligned}$$

Podemos concluir que

$$W_{mapS}(f,n) \in O(n)$$

`Profundidad`

$$\begin{aligned} S_{mapS}(f,n) &= S_{lengthA}(n) + S_{tabulateA}(f,n) \\ &= c_0 + \max_{i=0}^{n-1} S_f(a_i) \qquad\qquad S_f \in O(1) \\ &= c_0 + c_1 \end{aligned}$$

Podemos concluir que

$$S_{mapS}(f,n) \in O(1)$$

APPENDS

```
appendA :: A.Arr a -> A.Arr a -> A.Arr a
appendA a b = let l1 = lengthA a
                  l2 = lengthA b
                   l = l1 + l2
                   in tabulateA (\i -> if i<l1 then a!i else b!(i-l1)) l
```

Sea f

```
\i -> if i<l1 then a!i else b!(i-l1))
```

`Trabajo`

$$\begin{aligned} W_{appendS}(n_1,n_2) &= c_0 + W_{lengthS}(n_1) + W_{lengthS}(n_2) + W_{tabulateS}(f,n_1+n_2) \\ &= c_0 + c_1 + c_2 + \textstyle\sum_{i=0}^{(n_1+n_2)-1} W_f(a_i) \qquad (1) \end{aligned}$$

Ademas $W_f = c_3 + W_!$ y como $W_! \in O(1), W_f \in O(1)$

Entonces volviendo a (1) $\sum_{i=0}^{(n_1+n_2)-1} W_f(a_i) = c_3(n_1+n_2)$

Finalmente se tiene que $W_{tabulateS}(f,n_1+n_2) \leq c_4 + c_3(n_1+n_2)$ por lo que

$$W_{tabulateS}(f,n_1+n_2) \in O(n)$$

`Profundidad`

$$S_{appendS}(n_1,n_2) = c_0 + S_{lengthA}(n_1) + S_{lengthA}(n_2) + S_{tabulateA}(f,n_1+n_2)$$

Dado que lengthA $\in O(1)$ se tiene:
$S_{appendS}(n_1,n_2) = c_0 + \max_{i=0}^{(n_1+n_2)-1} S_f(a_i)$

Ademas $S_f(n) = c_0 + S_!(n) = c_0 + c_1$, entonces tenemos que $S_f(n) \in O(1)$

Retomando $S_{appendS}(n_1,n_2) = c_0 + \max_{i=0}^{(n_1+n_2)-1}(c_i) = c_0 + c_k$. Se concluye

$$S_{appendS}(n_1,n_2) \in O(1)$$

REDUCES

```
rdcAux ::(a -> a -> a)-> A.Arr a -> Int -> Int -> a
rdcAux f a i l = if i < div l 2
                 then f (a!(2*i)) (a!(2*i+1))
                 else a!(2*i)

reduceA :: (a -> a -> a) -> a -> A.Arr a -> a
reduceA f e a = let l = A.length a
                in case l of
                      0 -> e
                      1 -> f e (a!0)
              otherwise -> let xs = tabulateA (\i -> rdcAux f a i l) ((div l 2)+(mod l 2))
                           in reduceA  f e xs
```

Sea f'

```
\i -> rdcAux f a i l) ((div l 2)+(mod l 2))
```

$W_{f'}(i) = W_{rdcAux}(f,i) = c_0 + W_f(W_!(a,i), W_!(a,i)) =^* c_0 + W_f(a_{2i},a_{2i+1}) \quad {}^*W_! \in O(1)$
Luego tenemos que $W_f \in O(1)$, entonces, $W_{f'} \in O(1)$.
Ademas
$S_{f'}(i) = S_{rdcAux}(f,i) = c_0 + S_f(S_!(i_1), S_!(i_2)) =^* c_0 + S_f(c_1,c_2) = c_2 + S_f(c_3). \quad {}^*S_! \in O(1)$
Luego tenemos que $S_f \in O(1)$ por lo que $S_{f'} \in O(1)$.

`Trabajo`

$$\begin{aligned} W_{reduceA}(f,e,0) &= c_0 \\ W_{reduceA}(f,e,1) &= c_1 + W_f(e,a_0) \\ W_{reduceA}(f,e,n) &= c_2 + W_{lengthA}(n) + W_{tabulateA}(f',\lceil\tfrac{n}{2}\rceil)) + W_{reduceA}(f,e,\lceil\tfrac{n}{2}\rceil)) \\ &= k + W_{reduceA}(f,e,\lceil\tfrac{n}{2}\rceil)) + W_{tabulateA}(f',\lceil\tfrac{n}{2}\rceil)) \end{aligned}$$

Podemos ver que $W_{tabulateA}(f',n) = \sum_{i=0}^{n-1} W_{f'}(i) \qquad W_{f'} \in O(1)$

$$= \sum_{i=0}^{n-1} W_{f'}(i) \in O(n)$$

Por suavidad se tiene que

$$W_{reduceA}(f,e,n) \leq k + W_{reduceA}(f,\frac{n}{2}) + W_{tabulateA}(f',n)$$

Finalmente por caso 3 del **Teorema Maestro** tenemos que $W_{reduceA}(f,e,n) \in O(W_{tabulateA(f',n)})$ es decir:

$$W_{reduceA} \in O(n)$$

`Profundidad`

$$\begin{aligned} S_{reduceA}(f,e,0) &= c_0 \\ S_{reduceA}(f,e,1) &= c_1 + S_f(e,a_0) \\ S_{reduceA}(f,e,n) &= c_2 + S_{lengthA}(n) + S_{tabulateA}(f',\lceil\tfrac{n}{2}\rceil)) + S_{reduceA}(f,e,\lceil\tfrac{n}{2}\rceil) \\ &= k\ + \max_{i=0}^{\lceil\frac{n}{2}\rceil-1} S_{f'}(i) + S_{reduceA}(f,e,\lceil\tfrac{n}{2}\rceil) \qquad S_{f'} \in O(1) \\ &= k' + S_{reduceA}(f,e,\lceil\tfrac{n}{2}\rceil) \end{aligned}$$

En cada recursion la longitud de la lista se reduce a la mitad, se tiene que es del orden de $lg(n)$.
$S_{reduceA}(f,e,n) \in O(lg(n) + k')$ y se concluye que

$$S_{reduceA}(f,e,n) \in O(lg(n))$$

EXPANDS

```
expandA :: (a -> a -> a) -> A.Arr a -> A.Arr a -> A.Arr a
expandA f a a' = let l1 = lengthA a
                 in tabulateA (\i -> if (even i)
                                     then (a'!(div i 2))
                                     else (f (a'!(div i 2)) (a!(i-1)))) l1
```

sea f'

```
(\i -> if (even i) then (a'!(div i 2)) else (f (a'!(div i 2)) (a!(i-1)))
```

$W_{f'} \in O(1)$, ya que suponemos que $W_f, W_! \in O(1)$

`Trabajo`

$$\begin{aligned} W_{expandA}(f,n_1,n_2) &= c_0 + W_{lengthA}(n_1) + W_{tabulateA}(f',n_1) \\ &= c_0 + c_1 + \textstyle\sum_{i=0}^{n_1-1} W_{f'}(i) \\ &\leq c_0 + c_1 + nc_2 \end{aligned}$$

Finalmente tenemos que

$$W_{expandA} \in O(n)$$

`Profundidad`

$S_{f'} \in O(1)$, ya que suponemos que $S_f, S_! \in O(1)$

$$\begin{aligned} S_{expandA}(f,n_1,n_2) &= c_0 + S_{lengthA}(n_1) + S_{tabulateA}(f',n_1) \\ &= c_0 + c_1 + max_{i=0}^{n_1-1} S_{f'}(i), \quad S_{f'} \in O(1) \\ &\leq c_0 + c_1 + max_{i=0}^{n_1-1}(c_i) \\ &\leq c_0 + c_1 + c_3 \end{aligned}$$

Finalmente

$$S_{expandA} \in O(1)$$

SCANS

```
scanA :: (a -> a -> a) -> a -> A.Arr a -> (A.Arr a,a)
scanA f e a = let l = lengthA a
              in case l of
               0 -> (emptyA, e)
               1 -> (singletonA e, f e (a!0))
       otherwise -> let xs = tabulateA (\i -> rdcAux f a i l) ((div l 2)+(mod l 2))
                        (s,t) = scanA f e xs
                        in (expandA f a s, t)
```

`Trabajo`

$$\begin{aligned} W_{scanA}(f,e,0) &= c_0 \\ W_{scanA}(f,e,1) &= c_1 + W_f(e,a_0) \\ W_{scanA}(f,e,n) &= c_3 + W_{tabulateA}(f',n) + W_{scanA}(f,e,\lceil\tfrac{n}{2}\rceil) + W_{expandA}(f,n) \\ &= \textstyle\sum_{i=0}^{n-1} W_{f'}(a_i) + W_{scanA}(f,e,\lceil\tfrac{n}{2}\rceil) + W_{expandA}(f,n) \\ &\leq^* W_{scanA}(f,e,\lceil\tfrac{n}{2}\rceil) + n_1 + n_2 \end{aligned}$$

Entonces tenemos $W_{tabulateA}, W_{expandA} \in O(n)$

Al ser una función suave se tiene la recurrencia: $W_{scanA}(\frac{n}{2}) + n$ y
por el 3er caso del **Teorema Maestro** se concluye

$$W_{scanA}(n) \in O(n)$$

`Profundidad`

$$\begin{aligned} S_{scanA}(f,e,0) &= c_0 \\ S_{scanA}(f,e,1) &= c_1 + S_f(e,a_0) \\ S_{scanA}(f,e,n) &= c_3 + S_{tabulateA}(f',n) + S_{scanA}(f,e,\lceil\tfrac{n}{2}\rceil) + S_{expandA}(f,n) \end{aligned}$$

Tenemos que :
$S_{tabulateA}(f',n) = \max_{i=0}^{n-1}(S_{f'}(a_i)) =^* \max_{i=0}^{n-1}(a_i) = c_4$
$S_{expandA}(f,n) = c_5$

En cada recursion la longitud de la lista se reduce a la mitad, se tiene que es del orden de $lg(n)$.
$S_{scanA}(f,e,n) \in O(lg(n) + (c_4 + c_5))$ y se concluye que

$$S_{scanA}(f,e,n) \in O(lg(n))$$

$*\ W_{f'} \in O(1)$ demostrado en el ejercicio anterior "rdcAux" ya que asumimos $f \in O(1)$.