

CC3001

2018/04/12

Lista

Ejemplos: lista de cursos, de entos, de clientes de una empresa

① Abstract Data Type

② Estructuras de datos

- conjunto de datos

- acceder / primer elemento
/ proximo elemento (this)
(tamaño)

- modificar: agregar(x)
remover(x)

Arreglo

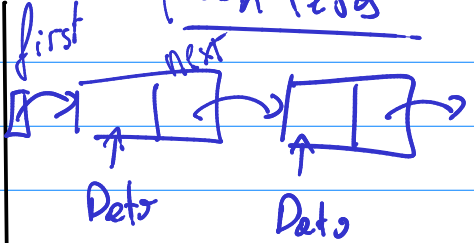
A: arreglo de datos
N: tamaño arreglo
n: cantidad de datos

$A[0]$; $i \leq 0$ $O(1)$
 $A[i]$, $i = i+1$ $O(1)$

$\begin{cases} \text{Si } n = N \text{ crear otro arreglo y copiar todos } O(n) \\ \text{Si } n < N \text{ } A[n] \leftarrow x \\ n = n+1 \end{cases}$

(-) buscar i t q $A[i] = x$
 $A[i] = A[n]$ $O(n)$
 $n = n-1$

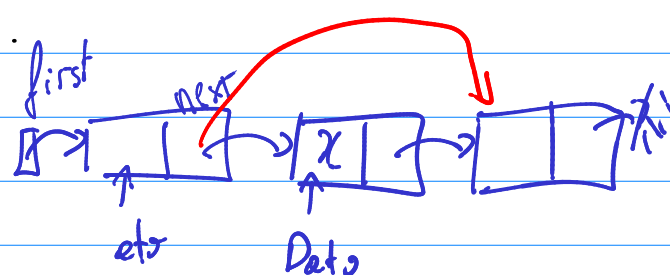
Puntos



return first $O(1)$
return this.next $O(1)$

$first = \text{new nodo}(x, first)$ $O(1)$

• buscar p buscando a
cedula con $O(n)$
(p.Next).Dato = x
p.Next = p.Next.Next



II Stacks / Piles / LIFO

① Abstract Data Type

Conjunto de Datos

$A: \text{Arreglo}[0..N-1]$

$n: \# \text{ elementos}$



n

isEmpty()

top()

isEmpty()

push(x)

Si $(n < N)$

$A[n] = x$; $n = n + 1$

Si $(n == N)$ crea otro arreglo
más grande y
copia todo $O(n)$

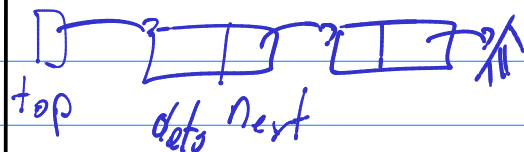
pull()

$n = n - 1$

$O(1)$

② Estructuras de datos

cedulas

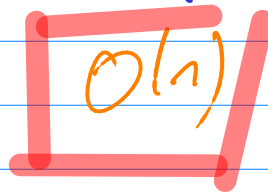


?

return top.datos $O(1)$

return (top == null) $O(1)$

top = new cedula(x, top)



top = top.next $O(1)$

III Fila / FIFO

LIFO Last In First Out → Pila
= FIFO → Fila
LIFO
FILO

① Abstract Data Type

|

② Estructuras de datos

|

IV Fila de prioridad

① Abstract Data Type

Conjunto de pares $(\text{Dato}, \text{Prioridad})$
 (x, p)

$\text{isEmpty}()$

$\text{add}(x, p)$

$\text{remove}(x)$

$\text{next}()$

↓
 $\text{min}()$

→ devuelve el elemento
de prioridad mas "alta"
en la Fila

y le saca de la lista