Laboratorio S07. AYED-02

Jefer Alexis González Romero

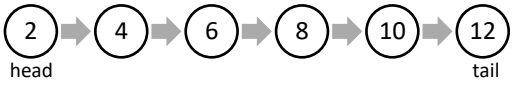
DOCUMENTO TÉCNICO

Implementación del método Delete para LinkedList

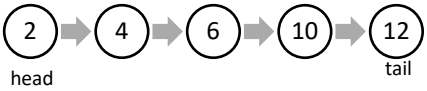
Requisitos

Especificación

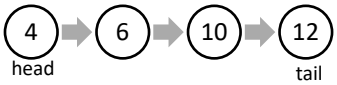
Definir el comportamiento delete_element(value). Elimina el nodo relacionado al valor (si existe) conservando la integridad referencial.



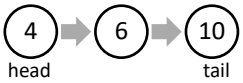
delete_element(8)



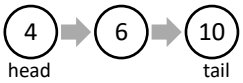
delete_element(2)



delete_element(12)



delete_element(14)



Entrada:

Se reciben los elementos que tendrá la lista enlazada separados por un espacio, en la siguiente línea se debe ingresar los valores (separados por espacios) que se desean eliminar de la lista entregada anteriormente.

Salida:

La lista enlazada sin los valores que se dieron para eliminarlos.

Diseño

Estrategia

Se busca el valor en la lista, si no está, no se hace nada y si al contrario si se encuentra, se evalúa dependiendo del caso que presente:

- El elemento a eliminar es el único en la lista, entonces está quedará vacía y su cabeza será None.
- Si el elemento es la cabeza, el que está a continuación de este será la nueva cabeza.
- El elemento es la cola, entonces el valor anterior a este tendrá como siguiente None.
- Por último, si el valor está entre otros dos, se buscará el que está previamente a este con una nueva función search_prev(value), a este se le colocará como siguiente el que estaba después del elemento a eliminar.

Adjunto el programa con nombre “Delete LinkedList”.

Casos prueba

Entrada	Justificación	Salida
2 4 6 8 10 12 8	Elemento que está entre otros dos	2 4 6 10 12
2 4 6 10 12 2	El elemento a eliminar es la cabeza de la lista	4 6 10 12
4 6 10 12 12	No hay otro elemento seguido al que se desea eliminar	4 6 10
4 6 10 14	El elemento no está en la lista enlazada	4 6 10

Análisis

Temporal

En el mejor de los casos $T(n) = \Omega(1)$

En el peor de los casos $T(n) = O(n)$

Código

Documentación

Dentro del código