**Assignment – Vlad: The Allocator**

Not about memory, not about allocation. This is a **linked list assignment.**

* A doubly circular linked list
  + All the nodes point both ways (making it doubly)
  + The “last node” points back to the first node, connecting the list and making it circular. (making it circular)
* To traverse:
  + Create a pointer to the first node
  + Create a current pointer, which you traverse through the circular linked list until you get back to the first node.

What people think its about

* Write functions that allocate memory?

Take a look at the sample code.

To remove a node in a doubly linked list???

Curr -> next -> prev = curr -> prev;

Curr -> prev -> next = curr ->next;

Free(curr);

How to swap two nodes?  
I.e. swap node 1 and 2 in the doubly linked list 0 -> <- 1 -> <- 2 -> <- 3 -> 4 -> X

Think about the edge cases, where it can go wrong, where you can get to a NULL etc.

Don’t be afraid to create new pointers to hold on to variables in the linked list.

**DO MORE PRACTISE WITH DOUBLY LINKED LISTS IN YOUR OWN TIME**

Pointer Arithmetic

* If you start off at the first pointer and you add +1 to the address
  + You move one data size across to the next element
  + E.g. for an int you move 4 bytes

Panthema.net/2013/sound-of-sorting/

Timo Bingmann Youtube