Data Structure Recitation 3.

1. Quiz

Name & netid.

```
class IntegerDoublyLinkedList {
    class Node {
        private int element;
        private Node next, prev;
        public Node getNext();
           "      "    getPrev();
        public void setNext(Node node);
           "      "    setPrev(Node node);
        public int getElement();
    }
    Node header, trailer
    int size;
}.
```

Q1: Implement "public void maxToFront()"

that 1. find the largest node and brings it to the first node.

2. maintain that header.getPrev() == null
   trailer.getNext() == null

3. if empty, do nothing.

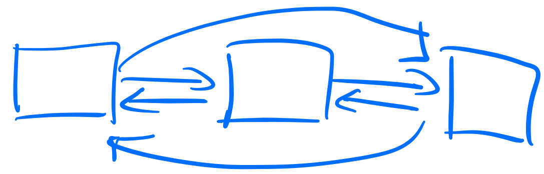
4. use only the methods specified.

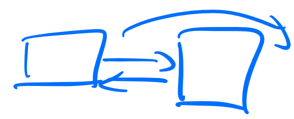
5. write java as authentic as you can, but don't stress too much on it.

(8pt) Try to have a good handwriting plz

Q2: If N is the size of the list, what's the worst case asymptotic complexity of this method? Explain your reasoning. (2pt).

A1 :



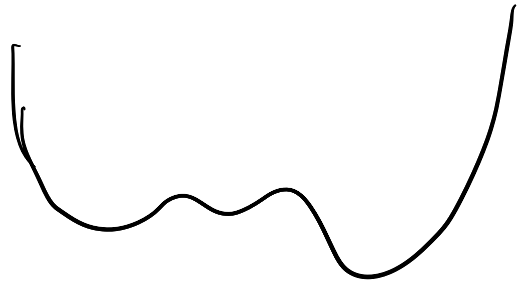


```
public void maxToFront() {
    if (size<2) return;
    Node node, max;
    node = header;  max=node;
    int m = header.getElement();
    while (node != null) {
        if (node.getElement() > m) {
            m = node.getElement();
            max = node;
        }.
        node = node.getNext();
    }.
    if (max == header)  return;
    max.getPrev().setNext(max.getNext());
    if (max != trailer) {.
        max.getNext().setPrev(max.getPrev());
    }.
    max.setNext(header);
    max.setPrev(null);
    header = max;
}.
```

A2: $O(N)$ because it needs to iterate through the whole linked list to find the maximum element.

2. Big-Oh notations.

$x^6 - 2x^5 + x^3$



$f(n) = O(g(n))$ if $\exists\, C, n_0 > 0$ such that.

if $n \geq n_0$, $f(n) \leq C g(n)$.

$f(n) = \Omega(g(n))$ if $\exists\, c, n_0 > 0$ such that

if $n \geq n_0$, $f(n) \geq c\, g(n)$.

$f(n) = \Theta(g(n))$ if $f(n) = O(g(n))$ and
$f(n) = \Omega(g(n))$.

Example

Show $x^6 - 2x^5 + x^3 = O(x^6)$.

$x = 2$, $C = 4$.

$$
\begin{aligned}
x^6 - 2x^5 + x^3 &\leq |x^6 - 2x^5 + x^3| \\
&\leq |x^6| + 2|x^5| + |x^3| \\
&\leq 4|x^6| = 4x^6.
\end{aligned}
$$

Show that $n\log n + n = O(n\log n)$.

pick $C=2$, $n_0 =$ the base of log.

Show that $(n+1)^5 = O(n^5)$

Show that $\log_a n = O(\log_b n)$

Show that $3^n \neq O(2^n)$.

suppose $\exists$ C and n_0 such that

if $n \geq n_0$, $3^n \leq C 2^n$

$\Rightarrow \frac{3^n}{2^n} = \left(\frac{3}{2}\right)^n \leq C \Rightarrow$ contradiction

Sufficient condition for Big-Oh notations

$$\lim_{n\to\infty} \frac{f(n)}{g(n)} \neq 0, \infty \Rightarrow f = \Theta(g)$$

$$\lim_{n\to\infty} \frac{f(n)}{g(n)} \neq \infty \Rightarrow f = O(g)$$

$$\lim_{n\to\infty} \frac{f(n)}{g(n)} \neq 0 \Rightarrow f = \Omega(g).$$

Show that $\log n = O(n^\alpha)$ $\forall \alpha > 0$.

$$\lim_{x\to\infty} \frac{\log x}{x^\alpha} = \lim_{x\to\infty} \frac{\frac{1}{x}}{\alpha x^{\alpha-1}}$$

$$= \lim_{x\to\infty} \frac{1}{\alpha} x^{-\alpha} = 0.$$

$$O(f(n)+g(n)) \stackrel{?}{=} O(\max\{f(n), g(n)\})$$

Yes

$d(n) = O(f(n))$, $e(n) = O(g(n))$

$$d(n) + e(n) \stackrel{?}{=} O(f(n)) + O(g(n))$$

$$d(n) - e(n) \stackrel{?}{=} O(f(n) - g(n))$$

No. $d(n) = e(n)$ is a counterexample

Order the following

$4n\log n + 2n$, 2^{10}, $2^{\log n}$,

$3n + 100\log n$, $4n$, 2^n, n^2+10n,

n^3, $n\log n$. $\log\log n$