## Wilson Lung (SE214

## Recitation 10

1. What is the worst case complexity of the following:

Just Move andres for curcular queve

| H of Front   | 1 |
|--|---|
| 10   | ( |
| <u> </u>   | , |
| 4 - 4 -  | 1 |
| H Front I  |   |
| $\Pi$ - $\Omega$ - $\Pi$ |   |
|  | ] |
| H Front J Front  | , |
| Ū-O-Ū-Ţ  | 1 |
| 1 0 0 TO 1   | 1 |
|  | 1 |
|  |   |
| <b>^</b> .   |   |
| Front  |   |

|   | Enqueue                   |      | Dequeue                   |            |
|---|---------------------------|------|---------------------------|------------|
| Array with pointer to next available slot   |                           | 0(1) | Shift array a<br>delete   | offer O(n) |
| Array without pointer to next available slot  | Find slot                 | 0(v) | Shift array               | (n)        |
| Circle Array  |                           | 0(1) |                           | 0(1) -     |
| Singly linked list with head reference and head as the front                        | Add Nav<br>Tail           | 0(n) | Remove                    | 0(1)       |
| Singly linked list with head and tail references with the head as the front         | Add New<br>Toul           | 0(1) | Remove<br>Head            | ()(1)      |
| Singly linked list with head and tail references with the tail as the front         | Add New<br>Head           | 0(1) | Remove<br>Tail            | 0(n)       |
| Doubly linked list with head and tail references with the head or tail as the front | Add New<br>Head DR<br>Tal | 6(1) | Remove<br>Head OR<br>Toul | (1)        |

2. Write a method called enqueue that accepts a *CircleArray* and an element to be added into the array. You can access the array, the front and the rear of the *CircleArray*.

public void enqueve (Circle Array arr, Object element)

if (arr. getRear()+1 % CAPACITY == arr. getFront())

throw new full Queue Exception ("Queue is Full")

else if (arr. getFront()==-1)

arr. getArray [0] : element

arr. setFront(0)

arr. setRear(0)

else

arr. setRear(larr. getRear()+1) & CAPACITY

arr. getArray [arr. getRear()] = element;

3. Given the function below, what is the result of fun (3,2)? 12, for (3,2) public int fun(int x, int y){ if(x == 0)return 😙 else } 4. Given the function below, how many total stars are printed as a result of stars What are the first 3 lines of output? public void stars(int n) { int i = 0; $\geqslant$  if (n > 1)stars(n-1);for(i = 0; i <  $\underline{n}$ ; i++) System.out.print("\*"); System.out.println();