StackADT (using DLinkedListADT)

class StackADT

```
     DLinkedListADT items // class attribute/member to store stack
items in linked list
     int capacity   // limit on how many elements there can be

     StackADT(int max)     // constructor to initialize list and
capacity
          if (max < = 0)
               throw exception
          Initialize items to empty list
          capacity = max

     boolean isEmpty() // return true if empty and otherwise false
          if items.count == 0
               return true
          return false

     boolean isFull() // return true if full and otherwise false
          if items.count == capacity
               return true
          return false

     void push(int item)  // add item to stack
          if items.count == capacity
               throw exception  // stack is full
          add item to the top/end of items

     int pop()  // remove item from stack
          if items.count == 0
               throw exception   // stack is empty
          return item from the top/end of the items

     int size()
          return number of elements in items
```

StackADT (using raw array)

class StackADT

```
     int top    // index of the top element in stack int
     items[]    // array to store stack items
     int capacity
```

```
    StackADT(int max)  // constructor
        if (max < = 0)
            throw exception

        Initialize array to max // programming language dependent
        capacity = max
        top = 0   // current slot for next item and also size

    boolean isEmpty()
        if top == 0  // no elements
            return true
        return false

    boolean isFull()
        if top == capacity
            return true
        return false

    void push(int item)
        if (isFull())
            throw exception
        add item to items[top]
        increment top


    int pop()
        if (isEmpty())
            throw exception

        decrement top
        return items[top]

    int size()
        return top
```

QueueADT (using DLinkedListADT)

```
class QueueADT

    DLinkedList items     // linked list to store items
    int capacity


    QueueADT(int max) // constructor to initialize empty linked list
        if (max < = 0)
```

```
                throw exception

          Initialize items to empty list
          capacity = max


     boolean isFull()
          if items.count == capacity //linked list is full
                return true
          return false

     boolean isEmpty()
          if items.count == 0 //linked list is empty
                return true
          return false


     void enQueue(int item)  // add to end/rear
          if (items.count == capacity)
                throw exception // queue full
          add item to the end/rear of the items


     int deQueue()  // remove from front/start
          if (isEmpty())
                throw exception
          return first element in items

     int size()
          return items.count
```

QueueADT (using raw array)

```
class QueueADT

     int end          // keep track of number of items and next slot
     int items[]      // array to store items
     int capacity


     QueueADT(int max)     // constructor to initialize queue to max
capacity
          Initialize array to hold max items
          end = 0
          capacity = max
```

boolean isEmpty()

```
if end == 0 // no elements in array/queue
     return true
return false
```

boolean isFull()

```
if end == capacity // max items in array
     return true
return false
```

void enQueue(int item)

```
if (isFull())
     throw exception
add item to items[end] // end/rear of items/queue
increment end
```

int deQueue()

```
     if (isEmpty())
          throw exception

     x = items[0]  // first item in queue
     loop to move all items to the previous index in array
          (e.g. x[0] = x[1])
     decrement end

     return x
```

int size()

```
     return end // number of items in array
```