Thursday, 30
June 2016

Crux

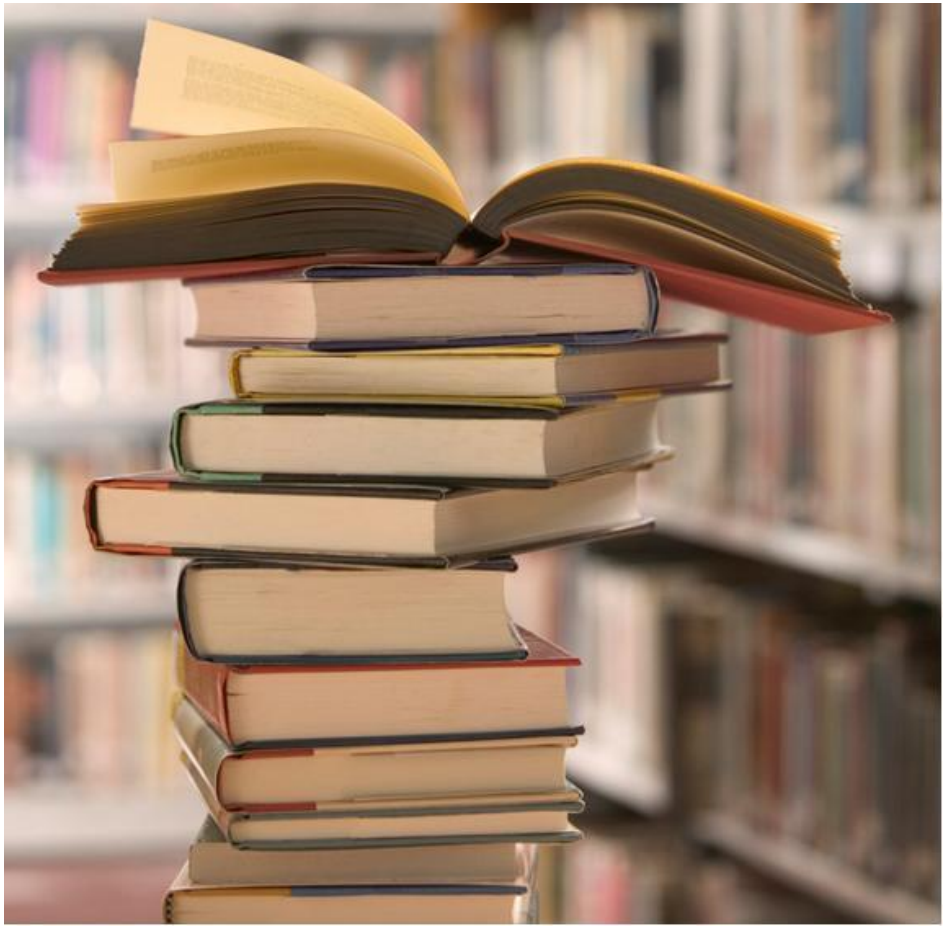
Data Structures -2

Stacks and Queues

Sumeet Malik

CODING
BLOCKS

Recursion and Pile of Books



CODING BLOCKS

Stacks

Stacks

```
class Stack{
    // accessor methods
    int size();
    boolean isEmpty();
    Object top() throws StackEmptyException;
    // update methods
    void push (Object element);
    Object pop() throws StackEmptyException;
}
```

How to implement Stack Class?

1. Arrays
2. Linked List

Lets Implement Our Own Stack Class Using Array

Homework : Implement Stack Class Using Dynamic Arrays

Your Turn: Implement Stack Class Using Linked List

Lets solve few problems

1. Given an expression check if brackets are balanced e.g. { a + [b+ (c + d)] + (e + f) }
2. Reverse a Stack with the help of another empty stack

Queues

Queue

```
class Queue{
    // accessor methods
    int size();
    boolean isEmpty();
    Object front() throws QueueEmptyException;
    // update methods
    void enqueue(Object element);
    Object dequeue() throws
    QueueEmptyException;
}
```

How to implement Queue Class?

1. Linked List
2. Arrays

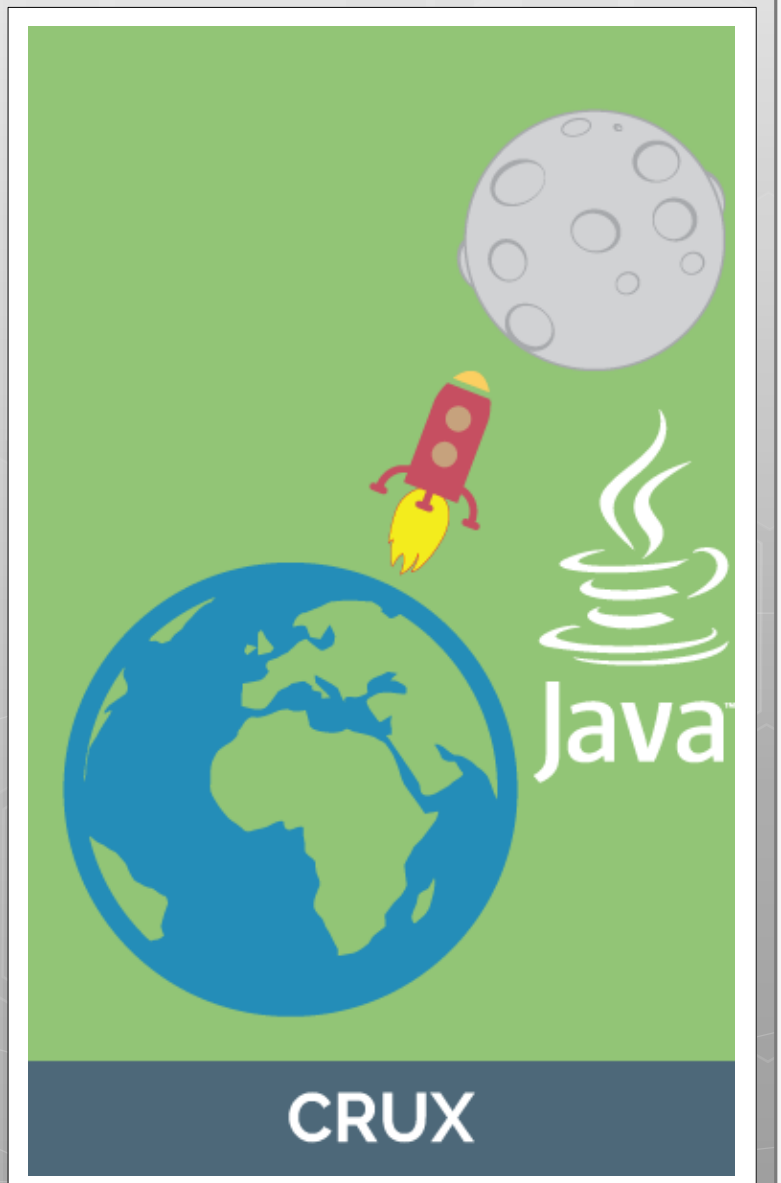
Lets Implement Our Own Queue Class Using Array

Homework : Implement Stack Class Using Dynamic Arrays

Your Turn: Implement Queue Class Using Linked List

Lets solve few problems

1. Reverse a Queue
2. Implement a Stack using Two Queues



Thank You!

Sumeet Malik

Sumeet.malik1188@gmail.com