Data Structure Lab (15CSE 281)
Lab Sheet I (20/07/16)

Instructions

- Use a text editor such as gedit to code your program and use terminal to type in the commands for compiling and executing the java program.
- Write the following code in a file Example.java (Your file name should be classname.java)

```
class Example {
public static void main(String [] args) {
System.out.println("This is a simple Java program.");
} }
```

- To compile the file, open your terminal and type javac Example.java
- To run the program java Example

Excercise 1 : Create a stack data structure and do the following operations push(x), pop(),getSize(),getTop(), isEmpty(), IsFull(), displayElements()

```
public class StackDemo {

    private static int n_size = 10;
    int A[] = new int[n_size];
    int top = -1;

    public void push(int x) {
        // Implement push operation
        //Display proper message
        // Call displayElements()
    }

     public void pop() {
        // Implement pop operation
        //Display proper message
        // Call displayElements()
     }

     public void displayElements() {
        for (int i = 0; i <= top; i++) {
        System.out.println(A[i]);
        }
      }
    public int getSize() {
```

```
}
public int getTop() {
}
public boolean isEmpty(){
}
public boolean isFull(){
}


public static void main(String[] args) {

  StackDemo s = new StackDemo();

s.push(5);

//Try push on a full stack

s.pop();

//Try pop on empty stack

int n= getSize();

//print n

int m= getTop();

}

}
```

Exercise 2: Implement the balanced parenthesis algorithm discussed in class as a function and add it into previous code. Input following expressions and check whether it is balanced or not.

{ ( ) { [( )] } }

[( ])

) (

{ ( ) ( ) ( ) [{ }] }