

# Data Structures (CPA250)

## *Lab03 - Stack Data Structure*

---

### 1 Stack data structure

Write code for Stack abstract data type ADT

Stack
top = [ ]
<pre>__init__(self) __contains__(self, item) __iter__(self) __str__(self)  push(self, item) pop(self) peek(self) isEmpty(self) size( self ) clear( self )</pre>

1. Object: Collection of data items such that last-in first out (LIFO) mechanism is maintained
2. Operations
  - push(x): adds an element x on the top of the stack
  - pop(): removes the top element of the stack. The next element will become the top element
  - isEmpty(): It returns true if the stack is empty, otherwise false
  - peek(): It returns the top element without removing it from the Stack
  - size(): It returns the number of items in the stack
  - display(): It displays all the elements stored in the stack
  - find(item): return true if item is found in the stack

---

## 2 Applications of Stack data structure

Write code for the following applications of Stack data structure

1. check parenthesis matching in an expression.
2. evaluate postfix expressions
3. convert infix form to postfix form
4. check brackets

StackApplications
<code>convertBase(self,num)</code>
<code>Infix2Postfix( self, expr )</code>
<code>evalPostfix(self, expr )</code>
<code>checkBrackets(self, statement)</code>