

CS218 - Data Structures
FAST NUCES Peshawar Campus
Dr. Nauman (recluze.net)

September 3, 2019

1 Circular Linked List in Python

Raster images of the notebook 06-circular-linked-list

Stack

A stack is essentially free in Python. Here's what happens when we use Python's list.

```
In [ ]: s = []  
  
s.append(12)  # append is the same as push  
s.append(5)  
s.append(55)  
  
print(s)  
  
print(s.pop())  
print(s.pop())  
print(s.pop())  
print(s.pop())  # <-- IndexError
```

```
In [ ]: s = []  
s.append(1)  
s.append(2)  
s.append(3)  
print(s)
```

```
In [ ]: s = Stack()
s.push(1)
s.push(2)
s.push(3)
# print(s[1])    # <--- error .... which, for us, is success since we have a stack and that's what we want.

# print(s.l[1])
# print(s.pop())
```

```
In [ ]: print(s.peak())
print(s.pop())
print(s.peak())
print(s.pop())
```

Case Study: Bracket Matching

```
In [ ]: a = '123'
b = '456'
dict(zip(a, b))
```

```
In [ ]: opening = '(['
closing = ')]'
mapping = dict(zip(opening, closing))
print(mapping)
mapping['{']
```

```
In [ ]: string = "[[]]{()}"
is_matched(string)
```

```
In [ ]: string = "2 + (3 * 5) * ((2 * 2) + 5)"
is_matched(string)
```

```
In [ ]: string = "2 + (3 * 5) * ((2 * 2) + 5) )"
is_matched(string)
```

Case Study: Binary to Decimal Conversion

```
In [ ]: def dec_to_bin(num):  
        s1 = []
```