

Lecture 14, 8 October 2024

Madhavan Mukund

<https://www.cmi.ac.in/~madhavan>

Programming and Data Structures with Python

User defined list

```
class List:
```

```
:
```

```
def append(-):
```

```
def insert(-):
```

```
def __str__(-):
```

```
def append(self, v):
```

```
if self.isempty():
```

```
self.value = v
```

```
return
```

loop till self.next == None

Iterative

Create a new node

Connect to this node

Recursive append

append-rec

if empty

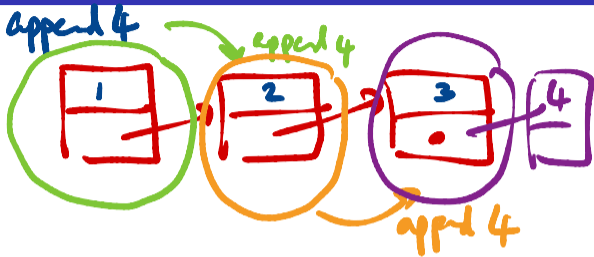
singleton

elif singleton

add a next node

else:

append to self.next



```
def appendr(self, v):
```

```
    if self.isempty():
```

```
        self.value = v
```

```
    elif self.next == None:
```

```
        n = List([v]) ← new node
```

```
        self.next = n
```

```
    else:
```

```
        self.next.appendr(v)
```



delete ?

def delete(self, v):



- remove first copy of v, if any
(No v - do nothing)

Cases

① Empty



return

②

Singleton



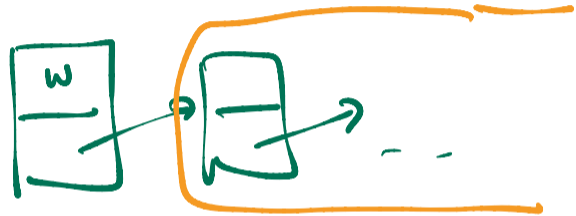
return

$w \neq v$



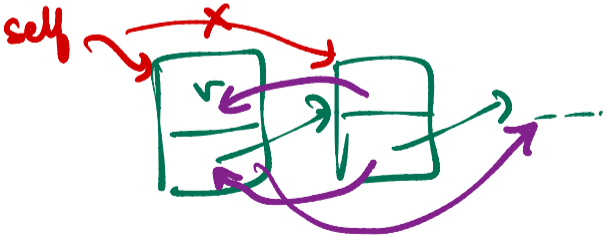
make empty
return

③ Non singleton

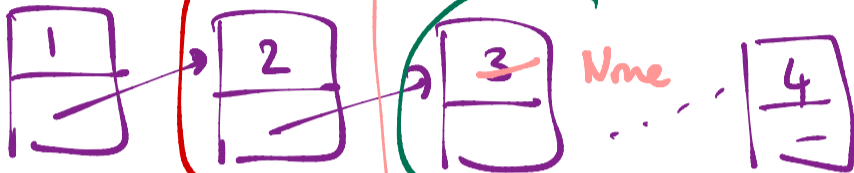


recursively
delete

if $self.next.value == None$
 $self.next = None$



delete 3

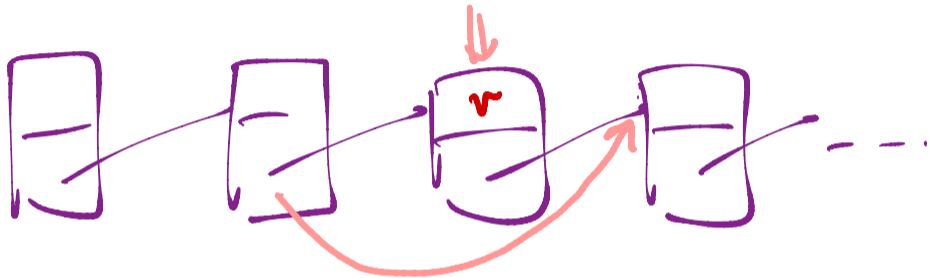


Not empty
Not singleton

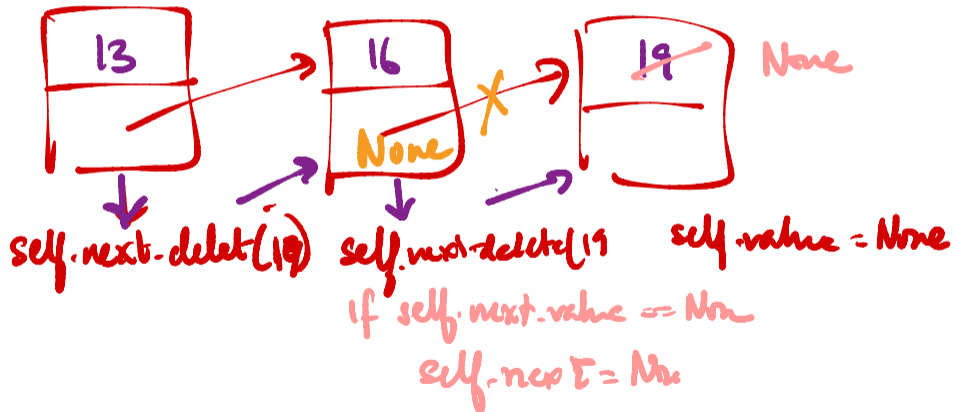
delete 3
Not Singleton
Not empty

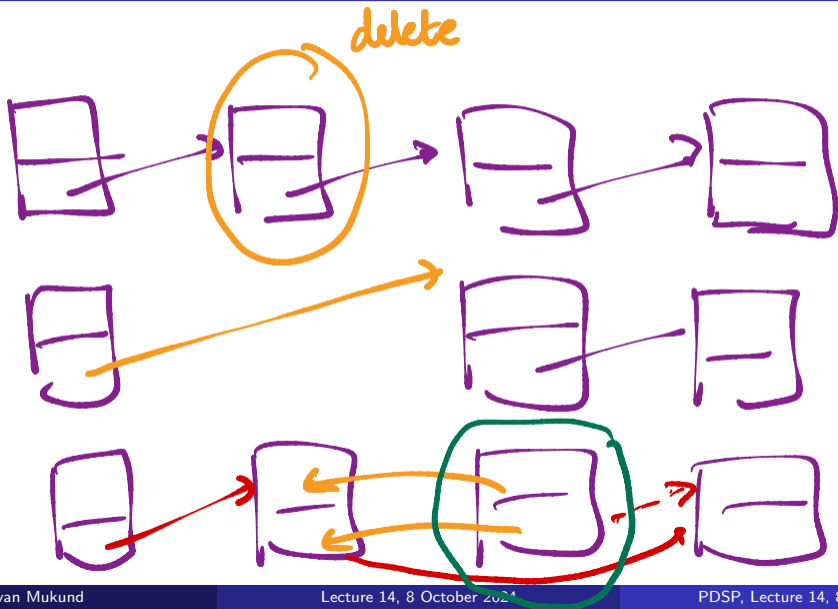
delete 3
Singleton!

Iterative delete



l. delete (19)





Errors

Name Error

Index Error

Key Error

Runtime Error

Value Error

⋮

Anticipate error and do something

Exception Handling




VS Normal

Recall

`int(s)` - works: if `s` represents a base 10 integer

- `ValueError`

Read a number (as a string) from keyboard
Not an int!



try:



Code that may generate an error

except:



What to do if error occurs

invalid input = True

While invalid input:

try:

read an input

convert to int

set invalid input to false

except:

EMPTY BLOCK

PASS

←

does nothing
avoids empty block

Input

```
x = input()
```

```
invalid = True
```

```
while (invalid) :
```

```
    try:
        xstr = input()
        xint = int(xstr)
        invalid = False
```

```
    except:
```

```
        pass
```

```
print(xint)
```

break

while (True):

try:

xstr = input()

xint = int(xstr)

break

except:

pass



except ValueError:

pass

try:

except ValueError:

except (IndexError, KeyError):

Counting matches per city

if city in d:

$d[\text{city}] += 1$

else:

$d[\text{city}] = 1$

try:

$d[\text{city}] += 1$

except KeyError:

$d[\text{city}] = 1$