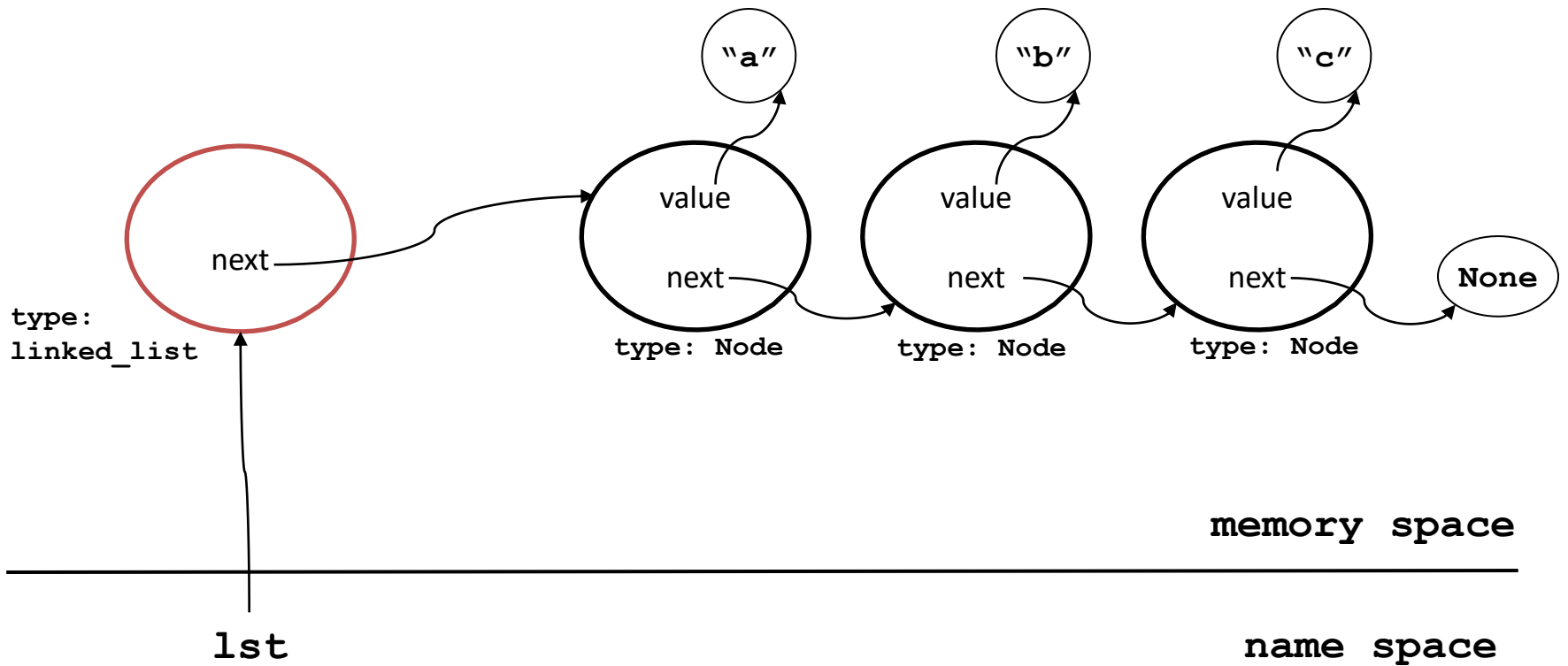
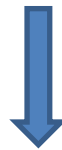
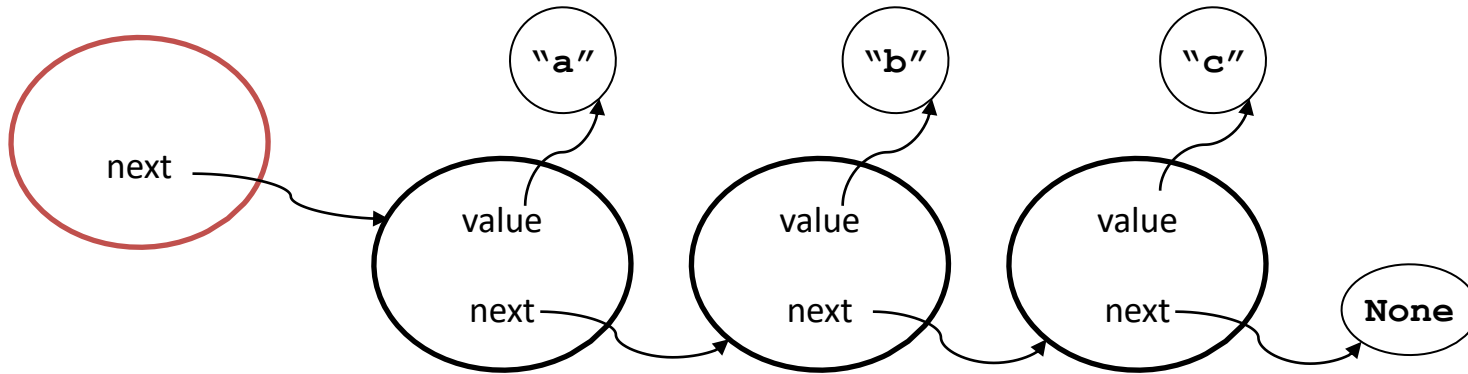


Linked lists memory view

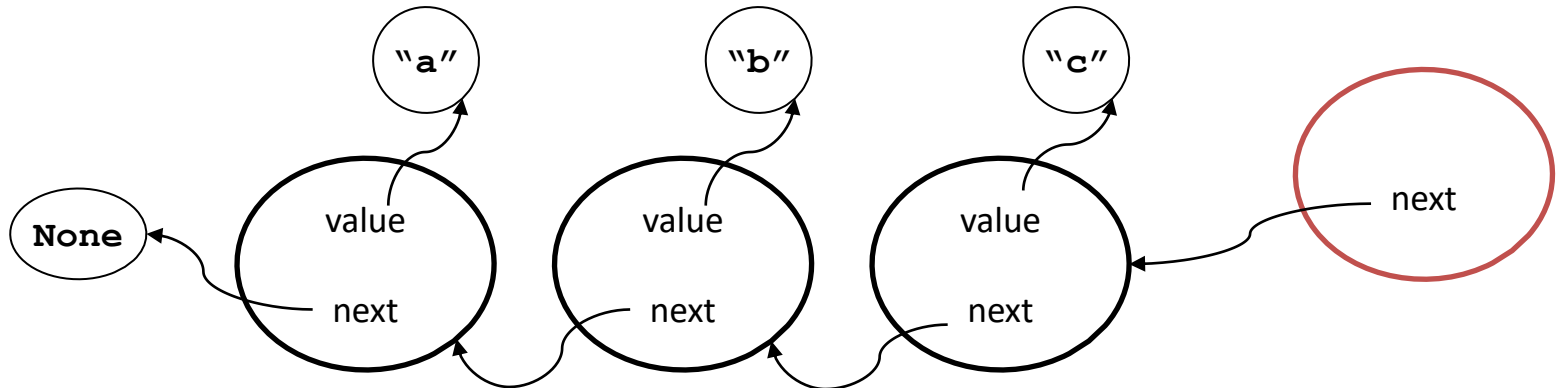
```
lst = Linked_list("abc")
```



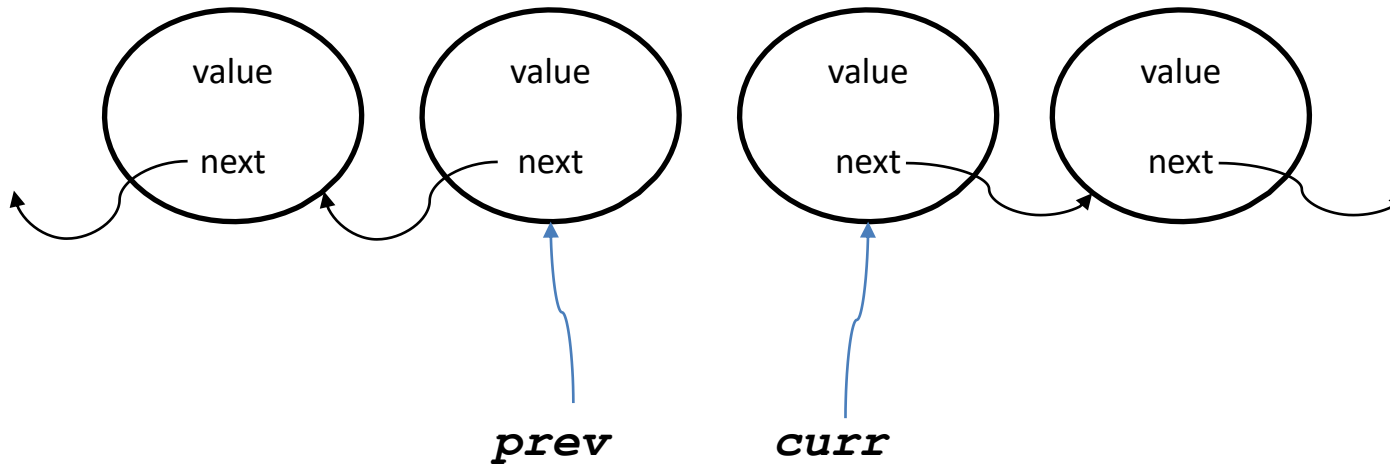
Exercise: implement list reverse with $O(1)$ additional memory



`lst.reverse()`

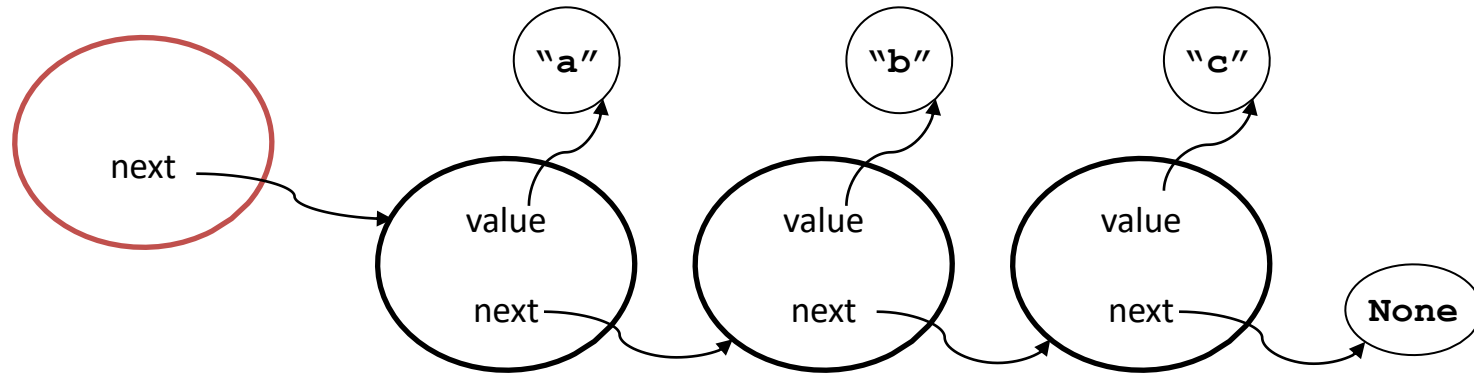


Guidance



1. Think about "the middle" of the process.
You have 2 consecutive elements *prev*, *curr*.
All elements up to *prev* were reversed.
Write the loop's body.
2. Think about the initialization of *prev*, *curr*
3. Think about termination of the loop and
extra actions after it

Solution

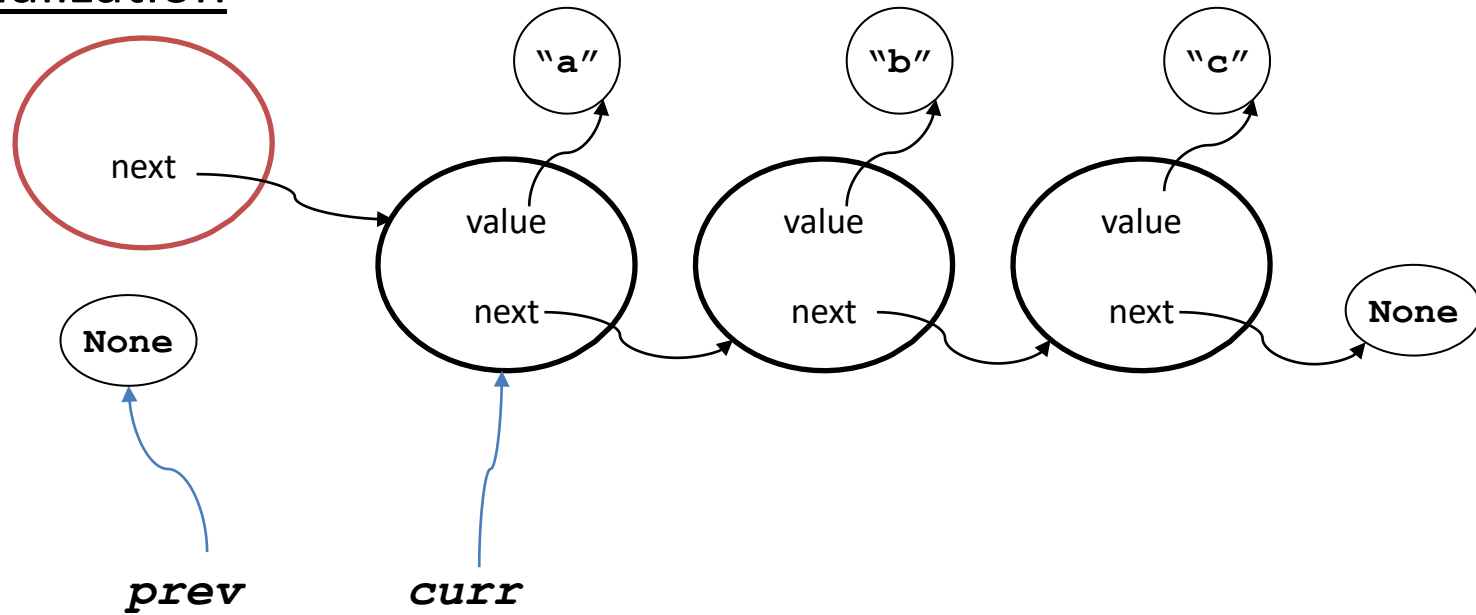


```
def reverse(self):  
    prev = None  
    curr = self.next  
    while curr != None:  
        tmp = curr.next  
        curr.next = prev  
        prev = curr  
        curr = tmp  
    self.next = prev
```

```
>>> lst = Linked_list("abcde")  
>>> lst  
[a,36067440] [b,36067376] [c,36067024] [d,36067312] [e,36066864]  
>>> lst.reverse()  
>>> lst  
[e,36066864] [d,36067312] [c,36067024] [b,36067376] [a,36067440]
```

Demo

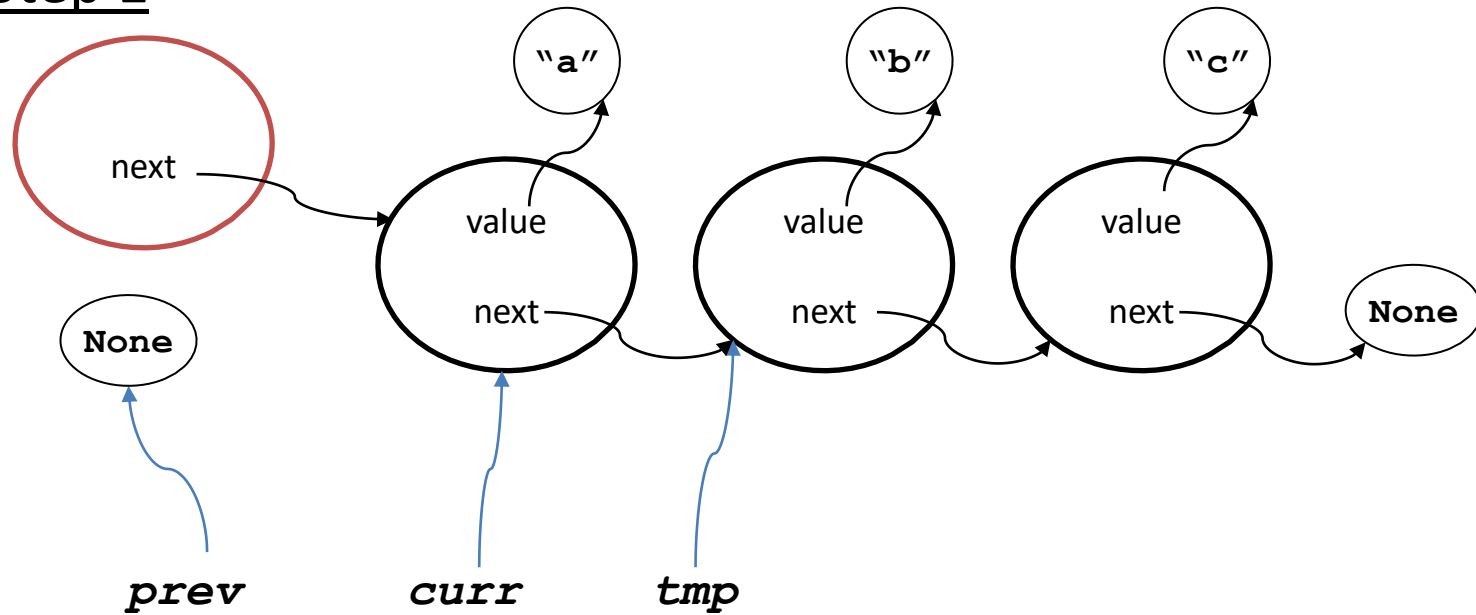
Initialization



```
def reverse(self):  
    prev = None  
    curr = self.next  
    while curr != None:  
        tmp = curr.next  
        curr.next = prev  
        prev = curr  
        curr = tmp  
    self.next = prev
```

Demo

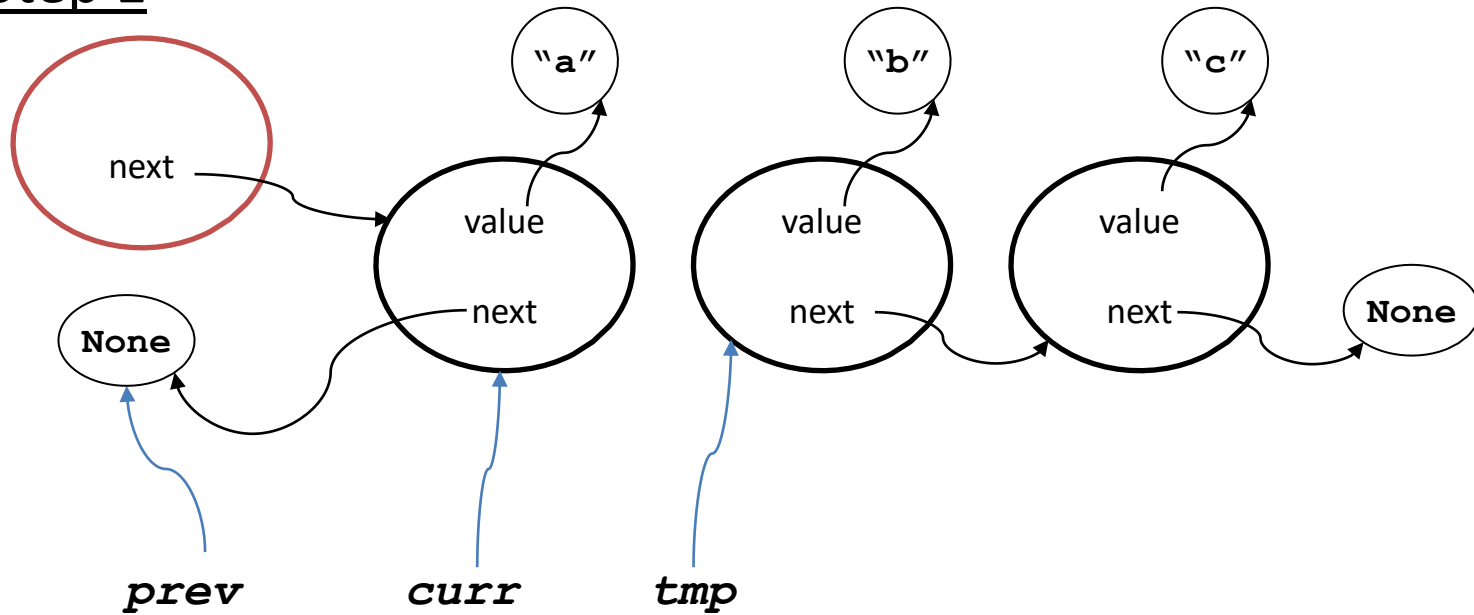
Step 1



```
def reverse(self):  
    prev = None  
    curr = self.next  
    while curr != None:  
        tmp = curr.next  
        curr.next = prev  
        prev = curr  
        curr = tmp  
    self.next = prev
```

Demo

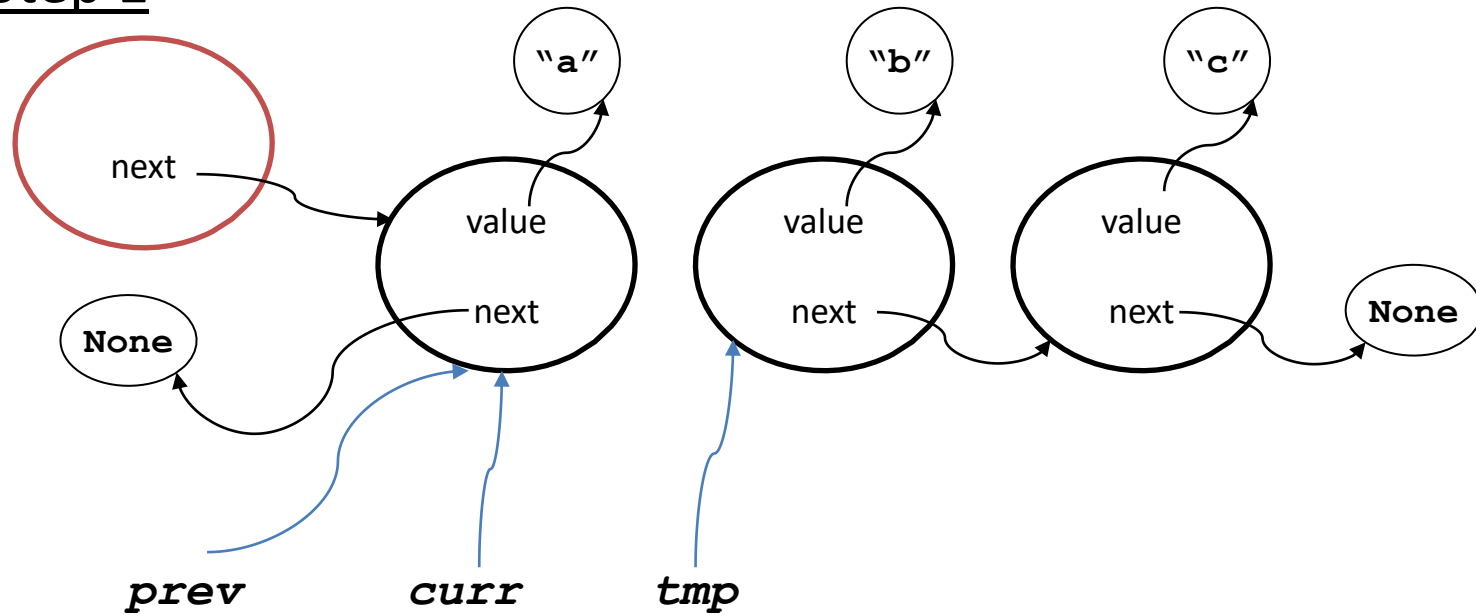
Step 1



```
def reverse(self):  
    prev = None  
    curr = self.next  
    while curr != None:  
        tmp = curr.next  
        curr.next = prev  
        prev = curr  
        curr = tmp  
    self.next = prev
```

Demo

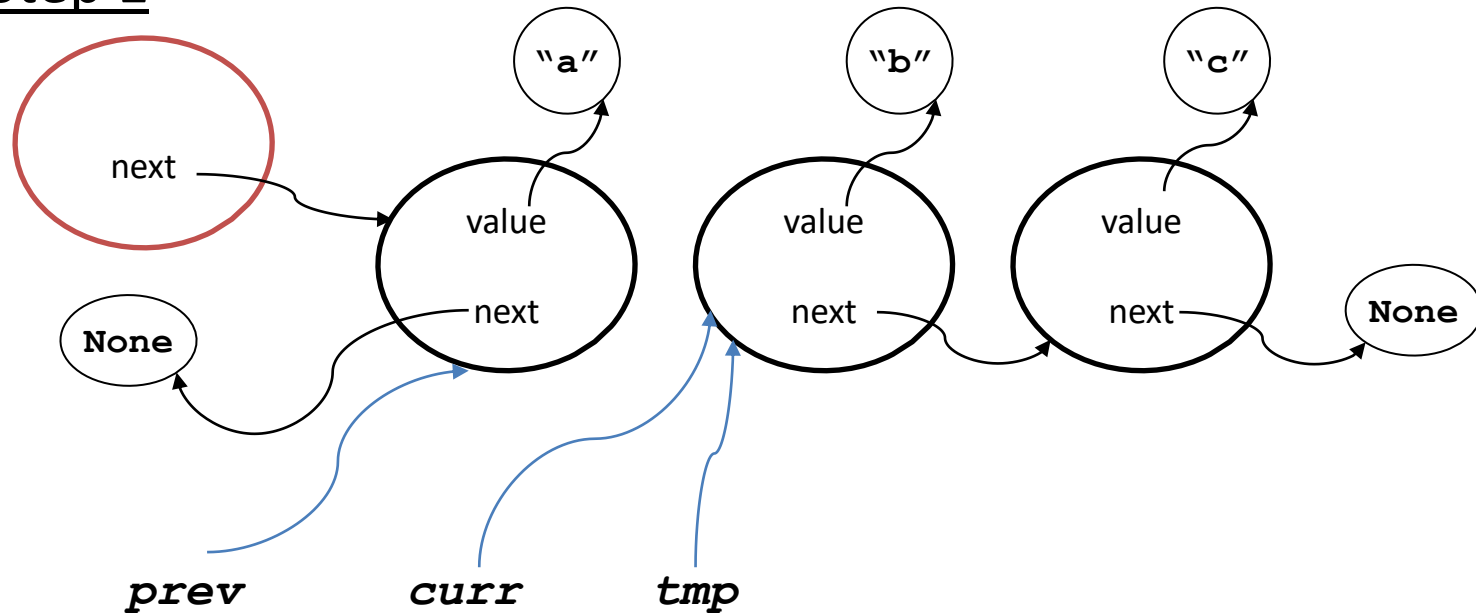
Step 1



```
def reverse(self):  
    prev = None  
    curr = self.next  
    while curr != None:  
        tmp = curr.next  
        curr.next = prev  
        prev = curr  
        curr = tmp  
    self.next = prev
```


Demo

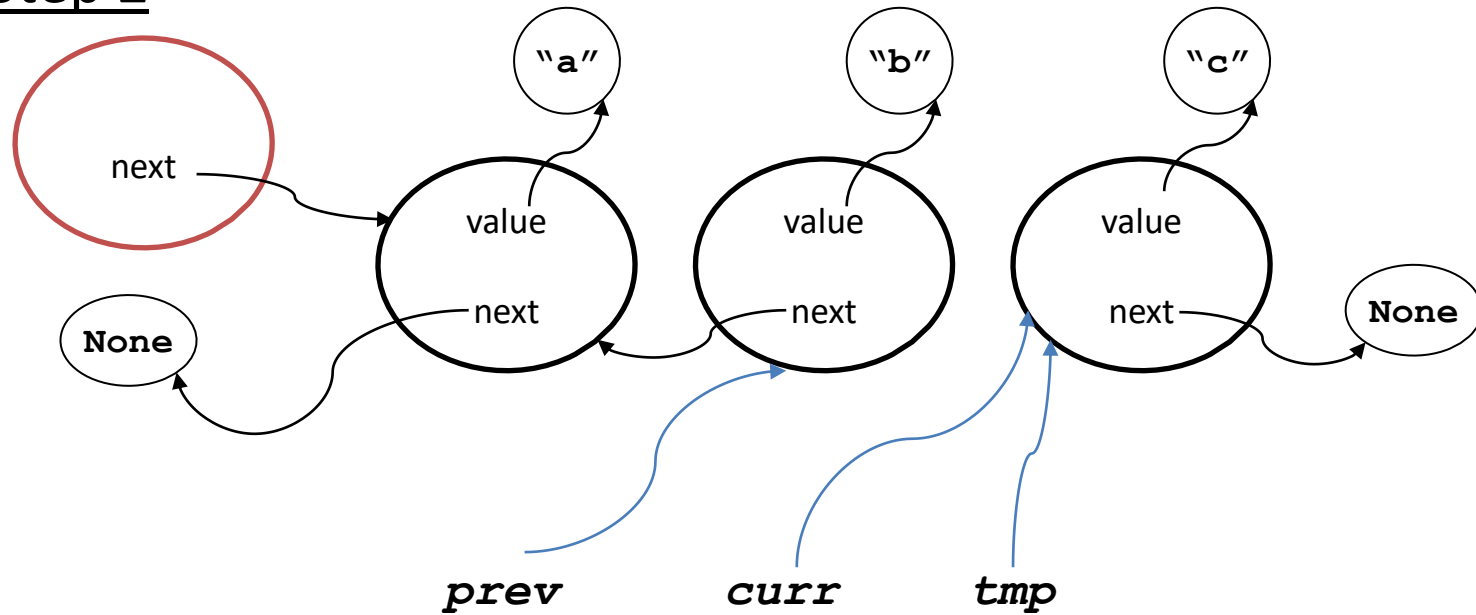
Step 1



```
def reverse(self):  
    prev = None  
    curr = self.next  
    while curr != None:  
        tmp = curr.next  
        curr.next = prev  
        prev = curr  
        curr = tmp  
    self.next = prev
```

Demo

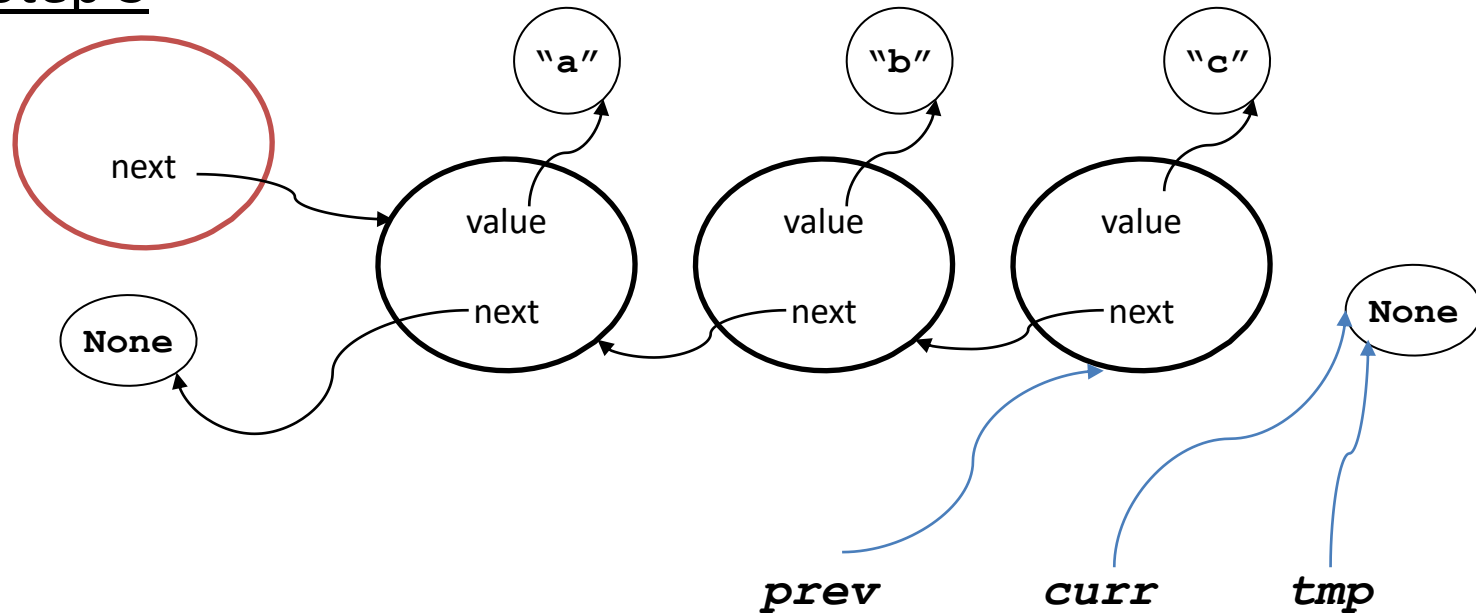
Step 2



```
def reverse(self):  
    prev = None  
    curr = self.next  
    while curr != None:  
        tmp = curr.next  
        curr.next = prev  
        prev = curr  
        curr = tmp  
    self.next = prev
```

Demo

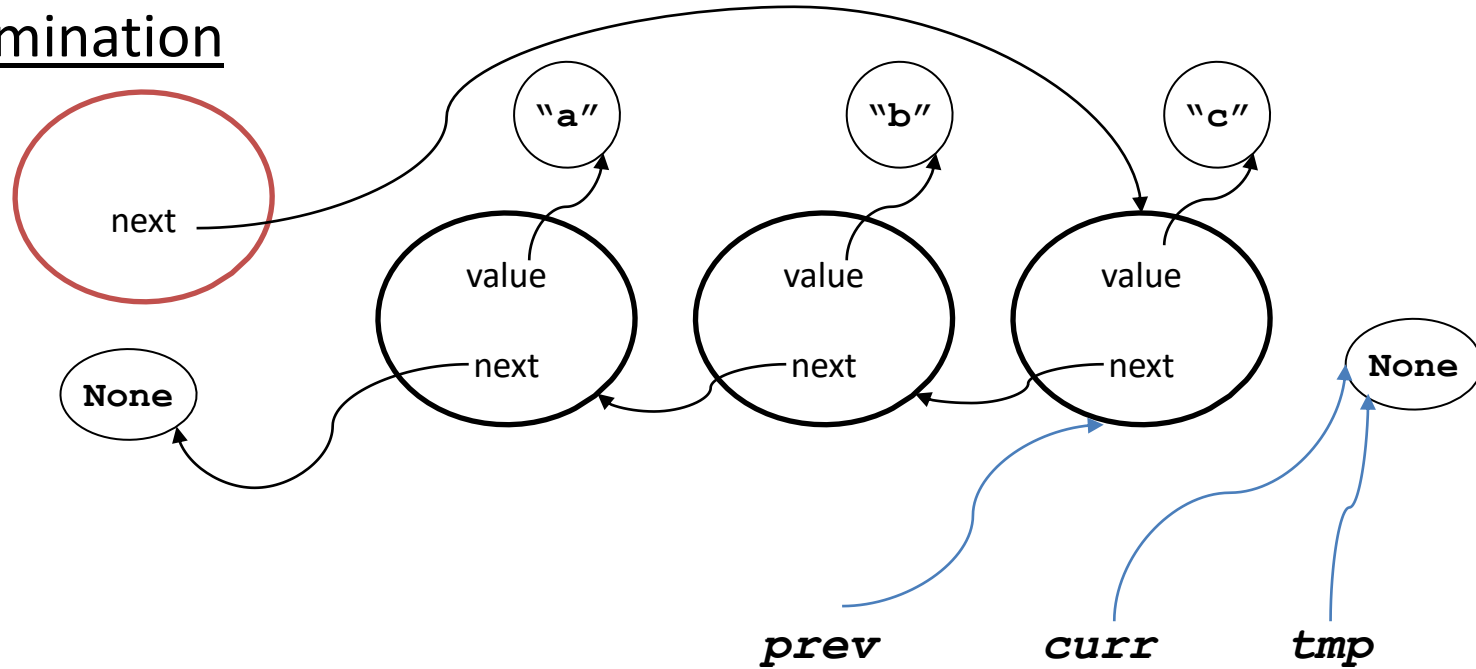
Step 3



```
def reverse(self):  
    prev = None  
    curr = self.next  
    while curr != None:  
        tmp = curr.next  
        curr.next = prev  
        prev = curr  
        curr = tmp  
    self.next = prev
```

Demo

Termination



```
def reverse(self):  
    prev = None  
    curr = self.next  
    while curr != None:  
        tmp = curr.next  
        curr.next = prev  
        prev = curr  
        curr = tmp  
    self.next = prev
```