HW6: Data Structure in Mathematics

Saen-Anan Bunyasiwa (SCIM 6105626)
Monchita Toopsuwan (SCIM 6105731)

February 18, 2020

R - 7.2

Describe a good algorithm for concatenating two singly linked lists L and M, given only references to the first node of each list, into a singly list L' than contains all the nodes of L followed by all the nodes of L followed by all the nodes of M.

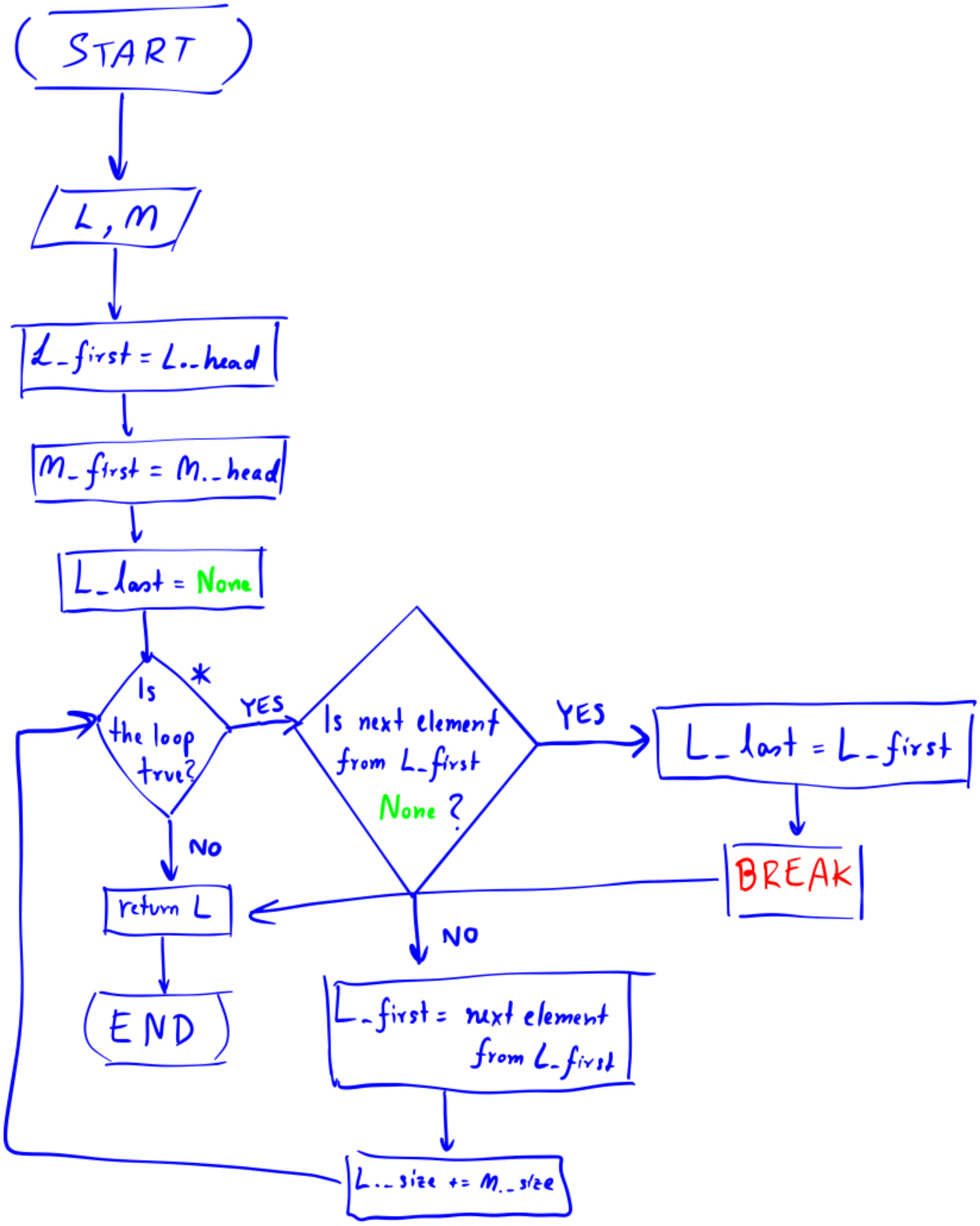
Solution Code

```
### Linked Stack Class goes here ###
def concat(L, M):
        L_first = L._head
        M_first = M._head
        L_last = None
        while True:
                if L_first._next is None:
                        L_last = L_first
                        break
                L_first = L_first._next
        L_last._next = M_first
        L._size += M._size
        return L

L = LinkedStack()
M = LinkedStack()
L.push(1)
L.push(2)
L.push(3)
M.push(4)
M.push(5)
M.push(6)
L_prime = concat(L, M)
for i in range(6):
print(L_prime.pop())
```

R - 7.2 Algorithms

Concatenation algorithms



* Note that the loop will run forever until breake

R - 7.2 Output

Solution Output

```
3
2
1
6
5
4
```

R - 7.5

Implement a function that counts the number of nodes in a circularly linked list.

Definition Code

```python
### CircularQueue Class goes here ###
def node_count(node, c = None):
        if node is c:
                return 0
        if c is None:
                c = node
        return 1 + node_count(node._next, c)
```

R - 7.5 Solution

Solution Code

```
S = CircularQueue()
S.enqueue(3)
S.enqueue(5)
S.enqueue(7)
S.enqueue(9)
S.enqueue(11)
S.enqueue(13)
print(S.dequeue())
print(S.dequeue())
S.rotate()
node = S._tail._next
print(node_count(node))
```

R - 7.5 Output

Solution Output

```
3
5
4 # Length of node
```

In - Class Homework

Use the python code `circular_queue.py` to create a linked list containing `3, 5, 7, 9, 11, 13`. Then, remove the first two elements.

In-Class Homework Solution

Solution Code

```python
if __name__ == '__main__':
        S = CircularQueue()
        S.enqueue(3)
        S.enqueue(5)
        S.enqueue(7)
        S.enqueue(9)
        S.enqueue(11)
        S.enqueue(13)
        print(S.dequeue())
        print(S.dequeue())
        print(S.first())
        print(S.tail())
        S.rotate()
        print(S.first())
        print(S.tail())
```

Solution Output

```
3
5
7
13
9
7
```