

Quiz

It's quiz time! Test yourself by solving these questions about doubly linked lists.

1

The code below is an implementation of the `Node` class for which of the following linked lists?

```
class Node:
    def __init__(self, data):
        self.data = data
        self.next = None
        self.prev = None
```

- ☐ A) Singly Linked Lists
- ☐ B) Circular Linked Lists
- ☐ C) Doubly Linked Lists
- ☐ D) None of the above

2

What is the output of the following code?

```
class DoublyLinkedList:
```

```
def delete(self, key):
    cur = self.head
    while cur:
        if cur.data == key:
            if cur.next:
                nxt = cur.next
                prev = cur.prev
                prev.next = nxt
                nxt.prev = prev
                cur.next = None
                cur.prev = None
                cur = None
                return
            else:
                prev = cur.prev
                prev.next = None
                cur.prev = None
                cur = None
                return
        cur = cur.next

dllist = DoublyLinkedList()
dllist.append(1)
dllist.append(2)
dllist.append(3)

dllist.delete(3)
dllist.delete(1)

dllist.print_list()
```

- ☐ A) 1 2 3
- ☐ B) 2
- ☐ C) No output
- ☐ D) Error

3

Doubly Linked List has more efficient methods to insert and delete elements than singly linked lists.

☐ A) True

☐ B) False

4

The `prev` attribute of a Node object is supposed to point to which of the following?

☐ A) The `next` attribute of the previous node in the doubly linked list

☐ B) The `data` attribute of the previous node in the doubly linked list

☐ C) The previous node in the doubly linked list

☐ D) The head node of the doubly linked list

5

What is the output of the following code?

```
class DoublyLinkedList:
```

```
class DoublyLinkedList:
    ...
    def print_list(self):
        cur = self.head
        while cur.next:
            cur = cur.next

        while cur:
            print(cur.data)
            cur = cur.prev

dllist = DoublyLinkedList()
dllist.prepend(0)
dllist.append(1)
dllist.append(2)
dllist.append(3)
dllist.append(4)
dllist.prepend(5)

dllist.print_list()
```

☐ A) 5 0 1 2 3 4

☐ B) 4 3 2 1 0 5

☐ C) No output

☐ D) Error

CHECK ANSWERS