

HƯỚNG DẪN THỰC HÀNH MÔN KIỂM TRA PHẦN MỀM TUẦN 4

GV: Nguyễn Hữu Phát

Cho chương trình thêm một nút vào danh sách liên kết đơn một chiều như sau

```
package w4_ktpm_linkedlist;

class Node {

    int data;
    Node next;

    Node(int d) {
        data = d;
        next = null;
    }
}
```

```
package w4_ktpm_linkedlist;

class LinkedList {

    Node head;

    // To add a new node at the front of the linked list, we create a new node
    // and make it the new head.
    public void push(int new_data) {
        Node new_node = new Node(new_data);
        new_node.next = head;
        head = new_node;
    }

    // Suppose we have a pointer to a node (prev_node), and we want to insert a
    // new node after it.
    public void insertAfter(Node prev_node, int new_data) {
        if (prev_node == null) {
            System.out.println("The given previous node cannot be null");
            return;
        }
        Node new_node = new Node(new_data);
        new_node.next = prev_node.next;
        prev_node.next = new_node;
    }
}
```

```

    // To add a new node at the end of the linked list, we traverse the list
    until we reach the last node and then append the new node.
    public void append(int new_data) {
        Node new_node = new Node(new_data);
        if (head == null) {
            head = new_node; // If the list is empty, make the new node the head
            return;
        }
        Node last = head;
        while (last.next != null) {
            last = last.next;
        }
        last.next = new_node;
    }

    // Print all nodes recursively
    public void printAll() {
        printNodes(head);
    }

    private void printNodes(Node current) {
        if (current == null) {
            return; // Base case: reached the end of the list
        }
        System.out.print(current.data + " ");
        printNodes(current.next); // Recurse to the next node
    }

    // Method to clear all nodes (set head to null)
    public void clear() {
        head = null;
    }
}

```

Yêu cầu

Xác định đoạn mã trên:

- Biến sử dụng chính đầu vào, ra.
- Các trường hợp nào có return? đảm bảo đúng?
- Phần nào là phần chính của giải thuật? và mục tiêu của nó?
- Đoạn mã nào đảm bảo sự kết thúc của danh sách liên kết đơn? có đúng đắn?

Viết lớp W4_KTPM_LinkedList bao gồm hàm main để thực thi các test-case sau nhằm kiểm tra sự chính xác của các hàm đã cho

- Danh sách ban đầu rỗng
- Danh sách có 1 phần tử là 7, thêm vào 4
- Danh sách có 2 phần tử là 5 -> 9, thêm vào 6
- Danh sách có 2 phần tử là 5 -> 9, thêm vào 12

Báo cáo

Báo cáo việc kiểm tra các test-case trên theo mẫu

ID	Input	Output expected	Output actual	Test result (passed/ not passed)

CHÚC THÀNH CÔNG