

You are to implement a *Stack* class with a *LinkedList* object as its underlying data structure. The *LinkedList* class header file is shown below. Please read the question carefully and only do what you are asked to – note that I am *not* asking you to implement the *LinkedList* methods described in the *LinkedList* header file that is shown below.

```
#include "Node.h"

class LinkedList{
private:
    Node* head;
    Node* tail;

    // PRE:
    // PARAM: current = node to be searched
    //         target = value to be searched for
    // POST: recursive helper method for LLsearch
    int searchHelper(Node* current, int target);

public:
    // Default (and only constructor)
    LinkedList();

    // Destructor
    ~LinkedList();

    // PRE:
    // POST: Returns 1 if list is empty, 0 otherwise
    int empty();

    // PRE:
    // PARAM: value = value to be appended
    // POST: Adds a new element to the tail of list
    void append(int value);

    // PRE:
    // PARAM: value = value to be inserted
    // POST: Adds a new element to the head of list
    void prepend(int value);

    // PRE: list is non-empty
    // POST: Removes element from head of list, and returns value
    int removeHead();

    // PRE: list is non-empty
    // POST: Removes element from tail of list, and returns value
    int removeTail();

    // PRE:
    // POST: prints list elements from head to tail
    void print();
};
```