

高级语言程序设计

上级报告3

习题5-3

题目

在JDK文档中查阅Stack接口，尝试用封装LinkedList的方式实现一个Stack容器。

思路

Just Code

```
class Stack<T> {
    private LinkedList<T> linkedList = new LinkedList<T>();
    public void push(T e) {
        linkedList.addLast(e);
    }
    public T pop() {
        return linkedList.pollLast();
    }
    public boolean empty() {
        return linkedList.isEmpty();
    }
    public int search(T e) {
        return linkedList.indexOf(e) + 1;
    }
    public T peek() {
        return linkedList.peekLast();
    }
    public String toString() {
        StringBuffer stringBuffer = new StringBuffer();
        for (T t : linkedList) {
            stringBuffer.append(t.toString() + ' ');
        }
        return stringBuffer.toString();
    }
}
```

习题5-4

题目

使用LinkedList实现一个类，该类有一个方法，该方法接受一个字符串作为参数，依次读取字符串中的字符，每次遇到“+”时就将“+”后面的一个字符压入栈中，每次遇到“-”时就将当前栈顶的字符弹出并在控制台打印，直至字符串遍历完，最后输出栈上剩余内容。实现main()方法调用这个方法，并输入字符串“+U+n+c---+e+r+t---+a-i-+n+t+y---++r+u---l+e+s---”，观察输出。

思路

根据题目编写代码，其内容如下：

```
public static void handle(String s) {
    LinkedList<Character> linkedList = new LinkedList<>();
    for (int i = 0; i < s.length(); ++i) {
        if (s.charAt(i) == '+') {
            linkedList.push(s.charAt(i + 1));
            i += 1;
        } else if (s.charAt(i) == '-') {
            System.out.print(linkedList.pop());
        }
    }
    while (!linkedList.isEmpty()) {
        System.out.print(linkedList.pop());
    }
}
```

程序输出

cnUtreaiytnursel-

习题5-7

题目

学习第8章中的文件输入/输出方法，从例5-6的TestMap.java文件中读入语句，并进行词频统计，进一步改写程序，统计TestMap.java中Java关键字的出现频率，在统计时，创建一个Set存储所有的Java关键字，在判断一个字符串记号是否为关键字时访问该Set。

思路

将文件读取内容作为字符串返回
统计所有单词的词频

输出结果

测试

```
public void test() {  
    String filename = "Test.java";  
    Map<String, Integer> res = new Count(filename).tokenize();  
    for (String s : res.keySet())  
        System.out.println(String.format("\'%s\': %d", s, res.get(s)));  
}
```

程序正常输出

```
new: 3  
static: 3  
if: 2  
public: 2  
class : 1  
else: 1  
for: 1  
import: 1  
int: 1 1  
null : 1  
this : 1  
true: 1  
void: 1  
while: 1
```