

# 数据结构 lab2

---

## 报告要求

---

1. 用简短的文字描述题目的做法和复杂度分析，不超过200字
2. 说明完成作业时获得了哪些人的何种帮助。

## 一，题目做法和复杂分析

---

### 卡片整理

This first problem was quite straightforward. I used an array of vectors. Each vector element represented a stack of cards. I used vector's push and pop function to remove and relocate cards. I achieved a status of accepted on all of the non-invisible test cases.

### 彩虹

For this problem, I used a simple naive (quite inefficient method). I looped through the vector, which stored the candies, found the relative maxes, and thus only saved the candies that were not to be eaten, then I would do the same process with the remaining candies.

### 烤馍片

I attempted this problem at first without the KMP method. Using brute force method, I searched the substrings of the text to find the longest starting prefix that was also a suffix. The brute force method however, exceeded the time limit on all the test cases. Then I implemented the KMP algorithm, of which allows us to find the greatest starting prefix that is also a suffix. It achieved accepted on 5 of the 10 test cases

### 树的遍历

In order to complete this problem, I first constructed a binary tree from the pre-order sequence (线序) given to us through recursion. The tree was constructed using `node` structs which I defined, which contained information about its value, or letter, and its left and right child. Then I traversed the tree using in-order (中序), printing the sequence in the correct order. I achieved a status of accepted on all of the non-invisible test cases.

## 二，帮助

---

I did all these problems by myself and only used class material as resources