

数据结构作业——线性表 (1)

基础部分

这部分练习为课堂讲授的内容和例题，要求全部理解并完成编码。

如果有能力，可以用 `class` 的形式完成对数据结构的定义。

- 1、 定义一个线性数据结构 `vector`，实现对 `vector` 操作的方法，包含
 - a. `vector` 的初始化并创建一个空表，
 - b. 清除 `vector` 中的所有元素，释放存储空间，使之成为一个空表
 - c. 返回 `vector` 当前的长度，若为空则返回 0
 - d. 判断 `vector` 是否为空，若为空则返回 1，否则返回 0
 - e. 返回 `vector` 中第 `pos` 个元素的值
 - f. 插入元素 `value` 到指定位置 `pos`
 - g. 删除指定位置 `pos` 元素
 - h. 按顺序输出整个 `vector` 所有的元素
 对 `vector` 的操作最好能实现空间的动态增长。
- 2、 实现一个双向链表，链表有头指针指向头结点，有尾指针指向尾结点，头结点和尾结点没有数据负载，作用仅为指向后继结点和前驱结点。实现对链表操作的方法，包括：
 - a. 插入值为 `value` 的节点到链表中，作为值为 `x` 的结点的后继结点
 - b. 删除链表中值为 `value` 的节点
 - c. 判断链表是否为空
 - d. 判断链表中是否有值为 `value` 的结点
 - e. 输出整个链表
 - f. 删除整个链表
- 3、 实现栈结构并实现对栈的入栈，出栈，取栈顶元素，判断栈是否为空的方法
- 4、 编写一个类，实现迎面增长栈，并实现两个栈的入栈，出栈，取栈顶元素，判断栈是否为空等方法的实现
- 5、 实现循环队列，并实现入队列，出队列，判断队列是否为空的方法。

以上面对各种线性数据结构的实现为基础，完成下面的任务：

- 6、 分别实现对向量和链表的插入排序算法
- 7、 实现对向量排序冒泡法和梳排序的算法
- 8、 实现对向量排序的 Shell 排序算法，分别用序列 1, 2, 4, 8, 2^n , ... 和 1, 3, 7, 15, 31, 63, 127, 255, ..., $2^j - 1$, ... 对向量进行排序。
- 9、 编写一个查找函数，对一个有序向量进行二分查找
- 10、 利用一个栈将中缀表达式转换为后缀表达式，并实现后缀表达式的计算

- 11、 分别利用和向量队列实现求解 Joseph 问题
- 12、 实现归并排序的函数，对向量进行排序
- 13、 实现快速排序的函数，对向量进行排序

完成下列算法练习：

- 21、 有一个有序数组 `array`，要求实现函数删除重复出现的数组元，使每个数组元最多出现两次，并返回数组的新长度。要求只能使用 $O(1)$ 的额外数组空间。
- 22、 有一个有序数组 `array`，要求实现函数从数组中找出两个数，相加之和等于目标数 `target`，如果数组中有多对数满足要求，找到一对数作为答案即可。请自己设计函数的原型，能使得调用者得到找到的两个数及其在数组中的位置。要求时间复杂度为 $O(n)$ ，只能使用 $O(1)$ 的额外数组空间。
- 23、 给定一个单链表，获得其所有的奇数节点，排在新链表的前半段，获得其所有偶数节点，排在新链表的后半段，保持奇数节点和偶数节点在原链表中的相对顺序。奇数节点和偶数节点指的是节点序号的奇偶性，不是节点值的奇偶性。要求算法使用 $O(1)$ 的额外数组空间，时间复杂度为 $O(n)$ 。
- 24、 有一个长度为 `n` 的整数数组，数组元在 $[0, 100]$ 区间内随机取值。请计算每一个元素后隔多少位置会有更大的值出现。(可以考虑用栈解决)如果后面没有更大的值，结果为 0。
比如，数组为 `[35, 36, 39, 30, 28, 16, 66, 33]`，答案为 `[1, 1, 4, 3, 2, 1, 0, 0]`