# Homework 8实验报告

twoSum实现：二重循环，依次遍历数组（第二级遍历比第一级初始位置+1），找到符合要求的target就把应用传入的a,b赋值，返回true，否则均赋值为-1，返回false。

TwoSum实现（用哈希表）：用map函数构建哈希表，key是两数之和，value是一个结构体，分别存两个加数的下标（从1开始，而且第一个加数在第二个加数的前面），用insert函数放入哈希表中，然后用find函数查找，当find返回map::end时表示未找到，函数返回false，否则就是找到了，find返回的first是key，second是value，把second传给引用location，返回true。

Sort类实现：创建链表，create函数根据输入的数组大小确定链表的长度以及依次建立链表，print函数从头节点Head开始依次打印各个节点，Print函数从给定节点开始一次打印各个节点（调试使用），主函数Mergesort归并排序链表，因为归并排序需要将节点全部拆散掉，所以需要将重新建立的链表赋值给Head，然后打印，调用mergeSortList函数负责从中间位置拆散各个节点，建立快指针和慢指针搜索之中位置，然后将其赋值给p1,p2，将p1的next赋值为nullptr，然后传入两段的起始位置递归调用mergeSortList，最后返回merge返回的指针。merge函数负责合并两个链表，建立结构体指针p,q，判断传入的节点的头位置谁小，小的为新的头位置，然后依次寻找两个链表中较小的下一个节点，为合并链表的下一个节点，一直到其中一个链表遍历至nullptr，然后将p指向剩余的各个节点的头位置（因为是链表，所以只要简单的将next指向就可以了，不需要像数组那样再逐个遍历），最后传回新建立链表的头指针q（p指针负责遍历链接节点）。