**1.实现下列返回一个向量中第k个最大元素的函数，1<= k <= n.**

int select\_kth\_biggest(vector<int> v, size\_t k);

①直接使用<algorithm>库里的sort函数,函数原型sort ( 数据起始, 数据末尾,排列方法 ) ; 排列方式不写默认从小到大排列,若想从大到小排,则

#include<algorithm>

bool compare(int a,int b) { return a>b; }

int a[10];

sort(a,a+10,compare);

//在这里就不需要对compare函数传入参数了，

//这是规则

②基于快排

选一个枢轴点，用快排的方法将数组分为两部分，位于枢轴点左边的数都比它大，位于枢轴点右边的数都比它小，

1）如果枢轴点的索引刚好是k-1,则此时它对应的就是数组的第k大的数；

2）如果比k-1大，那么第k大的数位于它的左边部分；

3）如果比k-1小，那么第k大的数位于它的右边部分；





**<二> 链表**

**1.查找链表倒数第k个值**

使用两个指针，第一个指针先走k-1步，然后第二个指针开始走。当第一个指针移动到最后时，第二个指针正好指向倒数第k个结点，只需要遍历一遍链表，显然更高效。

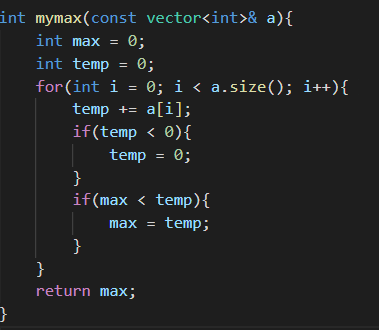
**2.删除中间节点**

实现一种算法，删除单向链表中间的某个节点（即不是第一个或最后一个节点），假定你只能访问该节点。

最大子序列和问题

寻找数组中子序列之和最大的那个值

1. 非循环数组



1. 可以循环的数组

转变思路：所有数之和 - 子序列最小和

