

# 数据结构和算法

作者: 小甲鱼

让编程改变世界

Change the world by program



## 单链表的整表删除

- 当我们不打算使用这个单链表时，我们需要把它销毁（真狠，不要就给别人嘛，还销毁~）。
- 其实也就是在内存中将它释放掉，以便于留出空间给其他程序或软件使用。
- 单链表整表删除的算法思路如下：
  - 声明结点p和q；
  - 将第一个结点赋值给p，下一结点赋值给q；
  - 循环执行释放p和将q赋值给p的操作；



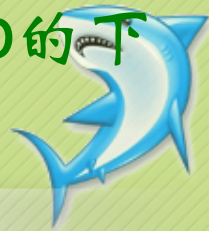
## 单链表的整表删除

- 我们说好的代码呢?
  - ClearList.c



## 单链表的整表删除

- 这段算法代码里，常见的错误就是有同学会觉得q变量没有存在的必要，只需要在循环体内直接写 `free(p); p = p->next;` 即可？
- 可这个世上没有无缘无故的爱，这样做会带来什么问题呢？
- 要知道p是一个结点，它除了有数据域，还有指针域。当我们做 `free(p);` 时候，其实是对它整个结点进行删除和内存释放的工作。而我们整表删除是需要一个个结点删除的，所以我们就需要q来记载p的下一个结点。这就好比\*\*\*\*\*。



# 单链表结构与顺序存储结构优缺点

- 我们分别从存储分配方式、时间性能、空间性能三方面来做对比。
- 存储分配方式:
  - 顺序存储结构用一段连续的存储单元依次存储线性表的数据元素。
  - 单链表采用链式存储结构, 用一组任意的存储单元存放线性表的元素。





# 单链表结构与顺序存储结构优缺点

- 时间性能:

- 查找

- 顺序存储结构 $O(1)$
    - 单链表 $O(n)$

- 插入和删除

- 顺序存储结构需要平均移动表长一半的元素, 时间为 $O(n)$
    - 单链表在计算出某位置的指针后, 插入和删除时间仅为 $O(1)$



# 单链表结构与顺序存储结构优缺点

- 空间性能：
  - 顺序存储结构需要预分配存储空间，分大了，容易造成空间浪费，分小了，容易发生溢出。
  - 单链表不需要分配存储空间，只要有就可以分配，元素个数也不受限制。
- 综上所述对比，我们得出一些经验性的结论：
  - 若线性表需要频繁查找，很少进行插入和删除操作时，宜采用顺序存储结构。
  - 若需要频繁插入和删除时，宜采用单链表结构。



## 单链表结构与顺序存储结构优缺点

- 比如说游戏开发中, 对于用户注册的个人信息, 除了注册时插入数据外, 绝大多数情况都是读取, 所以应该考虑用顺序存储结构。
- 而游戏中的玩家的武器或者装备列表, 随着玩家的游戏过程中, 可能会随时增加或删除, 此时再用顺序存储就不太合适了, 单链表结构就可以大展拳脚了。
- 当线性表中的元素个数变化较大或者根本不知道有多大时, 最好用单链表结构, 这样可以不需要考虑存储空间的大小问题。





## 单链表结构与顺序存储结构优缺点

- 而如果事先知道线性表的大致长度，比如一年12个月，一周就是星期一至星期日共七天，这种用顺序存储结构效率会高很多。
- 总之，线性表的顺序存储结构和单链表结构各有其优缺点，不能简单的说哪个好，哪个不好，需要根据实际情况，来综合平衡采用哪种数据结构更能满足和达到需求和性能。
- 最后，让我们通过欢乐的动画一起来回顾我们近期学过的知识吧~

