

计算机系统基础

Lab0 C-programming

姓名：傅文杰

学号：22300240028

2023 年 9 月 14 日

1 补全queue.h

在struct中增加成员变量，要求q_size和q_insert_tail函数在 $O(1)$ 的时间复杂度内完成 \Rightarrow 需要在读的过程中记录队列的长度和尾元素 \Rightarrow 尾指针list_ele_t *tail和队列长度size_t size需要成为成员变量

2 补全queue.c

2.1 NULL检查

1. 当为一个指针分配内存(malloc)时
2. 当函数中传入指针参数时

需要对pointer==NULL的情况进行特殊讨论

2.2 创建队列：queue_new

1. NULL检查(q)
2. 初始化成员变量(q->head,q->tail,q->size)

2.3 释放队列内存：`queue_free`

1. NULL检查(`q`)
2. 创建临时指针(`temp`)记录头指针
3. 更新头指针(`q->head = q->head ->next`)
4. 释放队列元素中指针的内存(`temp->value`)
5. 释放临时指针的内存(`temp`)

2.4 头(尾)插：`queue_insert_head(tail)`

1. NULL检查(`q,s,newh(newt),newh(newt)->value`)
2. 将`s`复制到新节点的`value`中(`strcpy(newh(newt)->value, s)`)
3. 更新队列(`q->head, q->tail, q->size`)

2.5 头删：`queue_remove_head`

1. NULL检查(`q, q->head`)
2. 创建临时指针(`temp`)记录头指针
3. 更新头指针
4. 复制`bufsize - 1`个字符到缓冲，注意加上`'\0'`
5. 释放临时变量的内存
6. 更新队列(`q->tail, q->size`)

2.6 求队列长度：`queue_size`

直接返回`q->size`

2.7 翻转队列: `queue_reverse`

1. NULL检查(`q, q->head`), 特殊情况(长度为1)不用翻转
2. 初始化三个指针: 前驱指针(`prev`), 当前指针(`current`), 后继指针(`next`)
3. 循环: 后继指针往前走, 当前指针的`next`指针回头指向前驱指针, 前驱指针和当前指针往前走
4. 头尾指针互换