lttzz / **Course-Design**

Branch: **master** **Course-Design** / T1 / **1.cpp** Find file Copy path

lttzz 简单的约瑟夫问题 3eb29fa 2 days ago

1 contributor

158 lines (140 sloc) 2.29 KB

```
/*************************************************************************
        > File Name:            1.cpp
        > File Category: Course Design
        > Author:               lttzz
        > Mail:                         3344517687@qq.com
 ************************************************************************/

#include <iostream>
using namespace std;

 //一开始想错了,以为 n 报数,最后会剩下 n-1 个人,事实上仍然只会剩下一个人
const int N = 13;
int L1 = 13, P = 0;
int del[N+5] = {0};

int L2 = 13;
struct Node
{
        int data;
        struct Node* next;
};
struct Node *head, *ptr, *tail;

void addOne();
void removeOne();
void show();
void solve();
void initList(int t);
void removeNode();
void showList();
void array();
void list();


int main(void)
{
        array();
        list();

        return 0;
}

/**
 * 编号有效加1
 */
void addOne()
{
        while (1)
        {
                P = (P + 1) % N;
                if (!del[P])
                {
                        break;
                }
        }
}

/**
 * 删掉某一个,并将剩余个数相应减1
 */
void removeOne()
{
```

```
63          del[P] = 1;
64          L1--;
65  }
66
67  void show()
68  {
69          for (int i = 0; i < N; i++)
70          {
71                  if (!del[i])
72                  {
73                          cout << i+1 << " ";
74                  }
75          }
76          cout << endl;
77  }
78
79  void solve()
80  {
81          while (L1 > 1)
82          {
83                  addOne();
84                  addOne();
85                  removeOne();
86
87                  addOne();
88          }
89  }
90
91  /**
92   * 初始化链表,构造有t个节点的链表,并将尾指针指向首结点
93   * @param t 节点个数
94   */
95  void initList(int t)
96  {
97          for (int i = 1; i <= t; i++)
98          {
99                  struct Node *newnode = new struct Node;
100                 newnode->data = i;
101                 newnode->next = NULL;
102                 if (NULL == head)
103                 {
104                         head = tail = newnode;
105                 }
106                 else
107                 {
108                         tail->next = newnode;
109                         tail = newnode;
110                 }
111         }
112         tail->next = head;
113 }
114
115 /**
116  * 删掉节点,只要剩余节点个数大于1,就继续删除下去
117  * 删除节点的时候顺便释放内存
118  * 删除完毕后将尾节点指向NULL,以便调用showList()函数
119  */
120 void removeNode()
121 {
122         struct Node *p = head;
123         struct Node *t = NULL;
124         while (L2 > 1)
125         {
126                 p = p->next;
127                 t = p->next;
128                 p->next = p->next->next;
129                 delete t;
130                 p = p->next;
131                 L2--;
132         }
133         head = p;
134         tail->next = NULL;
135 }
136
```

```
137 void showList()
138 {
139         for (ptr = head; ptr != NULL; ptr = ptr->next)
140         {
141                 cout << ptr->data << " ";
142         }
143         cout << endl;
144 }
145
146 void array()
147 {
148         solve();
149         show();
150 }
151
152 void list()
153 {
154         initList(N);
155         removeNode();
156         showList();
157 }
```