**基础训练**

程序名称：新建链表

link\_t \* init\_link()

{

link\_t \*link = (link\_t\*)malloc(sizeof(link\_t));

link->head = NULL;

link->length = 0;

return link;

}

程序名称：将指定元素放入到链表的末尾

int add\_link(link\_t \* link, element\_t data)

{

node\_t \*node = (node\_t\*)malloc(sizeof(node\_t));

memcpy(&node->data, &data, sizeof(data));

node->next = NULL;

if (node != NULL)

{

if (link->head == NULL)

{

link->head = node;

link->length++;

return 0;

}

node\_t \*p = link->head;

while (p->next != NULL)

{

p = p->next;

}

p->next = node;

link->length++;

return 0;

}

else

{

return -1;

}

return 0;

}

程序名称：将指定元素插入到链表的指定位置之前

int insert\_link(link\_t \* link, element\_t data, int location)

{

node\_t \*node = (node\_t\*)malloc(sizeof(node\_t));

if (node == NULL)

return -1;

memcpy(&node->data, &data, sizeof(data));

node->next = NULL;

node\_t \*p = link->head;

if (location == 1)

{

node->next = link->head;

link->head = node;

return 0;

}

for (int i = 1; i < location - 1; i++)

{

p = p->next;

}

node->next = p->next;

p->next = node;

link->length++;

return 0;

}

程序名称：将指定元素按照学号从小到大顺序插入到链表中

int insert\_link\_by\_order(link\_t \* link, element\_t data)

{

node\_t \*node = (node\_t\*)malloc(sizeof(node\_t));

if (node == NULL)

return -1;

memcpy(&node->data, &data, sizeof(data));

node->next = NULL;

if (data.stuID < link->head->data.stuID)

{

node->next = link->head;

link->head = node;

link->length++;

return 0;

}

node\_t \*p = link->head;

for (; data.stuID > p->next->data.stuID; p = p->next);

node->next = p->next;

p->next = node;

link->length++;

return 0;

}

程序名称：删除链表中指定姓名作为关键字的元素

int delete\_link(link\_t \* link, char \* name)

{

node\_t \*p = link->head;

if (strcmp(link->head->data.stuName, name) == 0)

{

link->head = link->head->next;

free(p);

link->length--;

return 0;

}

else

{

for (int i = 1; i < link->length - 1; i++)

{

if (strcmp(p->next->data.stuName, name) == 0)

{

node\_t \*q = p->next;

p->next = p->next->next;

free(q);

link->length--;

return 0;

}

p = p->next;

}

}

return -1;

}

程序名称：删除链表中总分小于某个指定值的所有元素

void delete\_link\_below(link\_t \* link, int x)

{

while (link->head->data.overall < x)

{

link->head = link->head->next;

link->length--;

}

int i = 0;

for (node\_t \*p = link->head; i < link->length - 1; i++)

{

if (p->next->data.overall < x)

{

p->next = p->next->next;

link->length--;

i--;

}

else

{

p = p->next;

}

}

return ;

}

程序运行结果：











