# 题目：链表练习

# 源代码：

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#include <malloc.h>

typedef struct students //定义一个链表

{

int judge;

int number;

char name[30];

char gender[10];

int grade;

int age;

struct students \*next;

}STU;

STU \*backward(STU \*head);

STU \*foreward(STU \*head);

void print(STU \*head);

void delet(STU \*head);

void change(STU \*head);

void insert(STU \*head);

int main()

{

STU \*head = NULL;

int judge\_direction = 0; //定义一个判断变量judge，用于判断输入方向；

int operate\_chosen; //定义一个判断变量operate\_chosen,用于判断操作方式；

printf("请输入按什么链表顺序进行输入：（“0”为正向，“1”为逆向）\t");

scanf("%d", &judge\_direction);

if (judge\_direction == 0)

{

head = foreward(head);

print(head);

}

else

{

head = backward(head);

print(head);

}

printf("请输入按什么方式进行对链表的操作：（“0”为删除节点，“1”为插入节点，“2”为修改链表节点值）\t");

scanf("%d", &operate\_chosen);

if (operate\_chosen == 0)

{

delet(head);

print(head);

}

else if(operate\_chosen == 1)

{

insert(head);

print(head);

}

else

{

change(head);

print(head);

}

return 0;

}

STU \*foreward(STU \*head) //正向输入；

{

STU \*new\_node; //定义一个过程结构体指针，指向新创建的结构体；

STU \*end; //定义链表尾；

while (1)

{

new\_node = (STU\*)malloc(sizeof(STU)); //为新建结构体分配内存；

printf("\n是否继续创建新的学生信息？ （输入“0”则结束创建，输入“1”则继续创建）\t");

scanf("%d", &new\_node->judge);

if (new\_node->judge == 0) //如果判断变量为0，则跳出循环；

break;

printf("学号："); //输入信息；

scanf("%d", &new\_node->number);

printf("姓名：");

scanf("%s", new\_node->name);

printf("性别：");

scanf("%s", new\_node->gender);

printf("年级：");

scanf("%d", &new\_node->grade);

printf("年龄：");

scanf("%d", &new\_node->age);

if (head == NULL) //如果链首为空；

{

head = new\_node; //链首指向新建结构体；

end = new\_node; //链尾也指向新建结构体；

new\_node->next = NULL;

}

else

{

new\_node->next = NULL;

end->next = new\_node;

end = new\_node;

}

}

return head;

}

STU \*backward(STU \*head) //逆向输入；

{

STU \*new\_node; //定义一个过程结构体指针，指向新创建的结构体；

while (1)

{

new\_node = (STU\*)malloc(sizeof(STU)); //为新建结构体分配内存；

printf("\n是否继续创建新的学生信息？ （输入“0”则结束创建，输入“1”则继续创建）\t");

scanf("%d", &new\_node->judge);

if (new\_node->judge == 0) //如果判断变量为0，则跳出循环；

break;

printf("学号："); //输入信息；

scanf("%d", &new\_node->number);

printf("姓名：");

scanf("%s", new\_node->name);

printf("性别：");

scanf("%s", new\_node->gender);

printf("年级：");

scanf("%d", &new\_node->grade);

printf("年龄：");

scanf("%d", &new\_node->age);

new\_node->next = head; //指针向前方传递；

head = new\_node;

}

return head;

}

void print(STU \*head) //输出打印；

{

STU \*p = head;

int i = 1;

printf("\n\n-------------------PRINT-------------------");

while (p!=NULL)

{

printf("\n节点序号：%d",i);

printf("\n学号：%d",p->number);

printf("\n姓名：%s",p->name);

printf("\n性别：%s",p->gender);

printf("\n年级：%d",p->grade);

printf("\n年龄：%d",p->age);

printf("\n");

p = p->next;

i++;

}

}

void delet(STU \*head) //删除节点

{

STU \*temp = head; //循环过程变量,暂时储存结构体；

STU \*delet\_element;

int i = 0; //循环过程变量；

int n; //要删除的链节序号；

printf("\n请输入您要删除的链节序号：");

scanf("%d", &n);

while (i < n && temp != NULL) //遍历所有的链节，找到要删除的一项，地址存入delet\_element；

{

delet\_element = temp;

temp = temp->next;

i++;

}

if (temp != NULL) //要删除的链节序号；

{

delet\_element->next = temp->next; //前节点的指针域越过要删除的节点指向下下个节点

free(temp); //释放要删除节点的内存；

}

else

{

printf("您要删除的节点不存在！");

}

}

void insert(STU \*head) //插入节点

{

STU \*temp = head; //循环过程变量,暂时储存结构体；

STU \*insert\_element;

int i = 0; //循环过程变量；

int n; //要删除的链节序号；

printf("\n请输入您要插入的链节序号：");

scanf("%d", &n);

while (i < n && temp != NULL) //遍历所有的链节，找到要插入的前一项，地址存入insert\_element；

{

temp = temp->next;

i++;

}

if (temp != NULL)

{

insert\_element = (STU\*)malloc(sizeof(STU));

printf("输入要插入的值\n");

printf("学号："); //输入信息；

scanf("%d", &insert\_element->number);

printf("姓名：");

scanf("%s", insert\_element->name);

printf("性别：");

scanf("%s", insert\_element->gender);

printf("年级：");

scanf("%d", &insert\_element->grade);

printf("年龄：");

scanf("%d", &insert\_element->age);

insert\_element->next = temp->next; //填充insert\_element节点的指针域，也就是说把insert\_element的指针域指向t的下一个节点

temp->next = insert\_element; //填充temp节点的指针域，把t的指针域重新指向insert\_element

}

else

{

printf("节点不存在");

}

}

void change(STU \*head) //修改链表节点值

{

STU \*temp = head; //循环过程变量,暂时储存结构体；

int i = 0; //循环过程变量；

int n; //要删除的链节序号；

printf("\n请输入您要修改的链节序号：");

scanf("%d", &n);

while (i < n && temp != NULL) //遍历所有的链节，找到要修改的一项；

{

temp = temp->next;

i++;

}

if (temp != NULL)

{

printf("\n输入要修改的值\n");

printf("学号："); //输入信息；

scanf("%d", &temp->number);

printf("姓名：");

scanf("%s", temp->name);

printf("性别：");

scanf("%s", temp->gender);

printf("年级：");

scanf("%d", &temp->grade);

printf("年龄：");

scanf("%d", &temp->age);

}

else

{

puts("节点不存在");

}

}

# 输出结果：

