1. 输入字符a至f,按输入字符的顺序建立一个字符结点（每个结点存储一个字符）的单向链表，输出该链表中的字符,并释放链表结点所占用的空间。

#include<stdio.h>

#include <malloc.h>

#include <stdlib.h>

typedef struct word

{

char c;

struct word\* next;

}word;

/\*create a chain table\*/

word\* creatnode(void)

{

int i;

char ch;//临时存放用户输入的结点的值

//分配一个不存放有效数据的头节点

word\* phead = (word\*)malloc(sizeof(word));

if (phead == NULL)

{

printf("分配失败\n");

exit(-1);

}

word\* ptail = phead;

ptail->next = NULL;

//建立链表

for (i = 0; i < 6; i++)

{

printf("请输入第%d个节点的值：", i + 1);

scanf("%c", &ch);

getchar();

word\* new = (word\*)malloc(sizeof(word));

if (new == NULL)

{

printf("分配失败\n");

exit(-1);

}

new->c = ch;

new->next = NULL;

ptail->next = new;

ptail = new;

}

return phead;

}

/\*print the chain table\*/

void translist(word\* phead)

{

word\* p = phead->next;

while (p != NULL)

{

printf("%c\n", p->c);

p = p->next;

}

}

/\*release memory\*/

void freep(word\* p)

{

word\* head = p;

word\* tmp = p;

while (tmp)

{

head = tmp;

tmp = tmp->next;

free(head);

}

}

int main()

{

word\* phead = NULL;//the head node

phead = creatnode();//create the chain table

translist(phead);//print the chain table

free(phead);//release memory

return 0;

}



2、创建一个 单向链表 来记录学生信息，人数3--5人；链表结点为结构变量，结构的要求如下：

struct stu\_info

{

         char stu\_num[10];  //学号

         char stu\_name[8];  //姓名

         char stu\_sex[2];    //性别

         int stu\_score;    //成绩

         struct stu\_info \*next;

};

程序设计要求：

（1）插入新的学生信息(插入节点的位置可任意指定）

（2）删除指定的学生信息

（3）根据学号查询并显示查询到的学生信息

（4）以上三项任务分别自定义函数实现，执行后显示执行结果

（5）程序运行后要求可以循环执行前三项操作，直到选择退出时结束程序

#include<stdio.h>

#include<stdlib.h>

#include<malloc.h>

#include<string.h>

/\*定义结构体\*/

typedef struct stu\_info

{

char num[10];//学号

char name[8];//姓名

char sex[2];//性别

int score;//成绩

struct stu\_info\* next;

}stu;

stu\* creat(int );//创建

void insert(stu\* );//插入

void delete(stu\* );//删除

void search(stu\* );//查找

int main()

{

int n;

printf("/\*学生信息管理系统\*/\n");

printf("请输入学生个数：");

scanf("%d", &n);

getchar();

/\*创建链表\*/

stu\* phead = creat(n);

/\*键入ctrl+Z终止程序\*/

printf("若要继续，请按回车\n");

char c1,c2,c3;

while ((getchar()) != EOF)

{

/\*插入\*/

printf("是否要插入信息？（y/n）");

scanf("%c", &c1);

getchar();

if (c1 == 'y')

{

insert(phead, n);

n++;

}

/\*删除\*/

printf("是否要删除信息？（y/n）");

scanf("%c", &c2);

getchar();

if (c2 == 'y')

{

delete(phead);

n--;

}

/\*查询\*/

printf("是否要查询信息？（y/n）");

scanf("%c", &c3);

getchar();

if (c3 == 'y')

search(phead);

printf("键入Ctrl + Z终止程序\n若要继续，请按回车\n");

}

/\*释放内存\*/

stu\* tmp=phead;

stu\* temp = phead;

while (temp)

{

tmp = temp;

temp = temp->next;

free(tmp);

}

return 0;

}

/\*创建链表\*/

stu\* creat(int n)

{

int i = 0;

char num[10];//学号

char name[8];//姓名

char sex[2];//性别

int score;//成绩

//分配头节点

stu\* phead = (stu\*)malloc(sizeof(stu));

if (phead == NULL)

{

printf("分配失败\n");

exit(-1);

}

stu\* ptail = phead;

ptail->next = NULL;

//创建链表

for (i = 0; i < n; i++)

{

printf("请输入第%d个学生的信息：\n", i + 1);

printf("学号：");

gets(num);

printf("姓名：");

gets(name);

printf("性别：");

gets(sex);

printf("成绩：");

scanf("%d", &score);

getchar();

stu\* pnew = (stu\*)malloc(sizeof(stu));

if (pnew == NULL)

{

printf("分配失败\n");

exit(-1);

}

strcpy(pnew->num, num);//学号

strcpy(pnew->name, name);//姓名

strcpy(pnew->sex, sex);//性别

pnew->score = score;//成绩

pnew->next = NULL;

ptail->next = pnew;

ptail = pnew;

}

return phead;

}

/\*插入信息\*/

void insert(stu\* phead, int m)

{

int i = 1;

int n;//节点位置

printf("请输入要插入的位置：");

scanf("%d", &n);

getchar();

stu\* tmp = phead;

stu\* temp = phead->next;

stu\* pnew = (stu\*)malloc(sizeof(stu));

char num[10];//学号

char name[8];//姓名

char sex[2];//性别

int score;//成绩

if (n < m)

{

for (i = 1; i < n; i++)

{

tmp = tmp->next;

temp = tmp->next;

}

printf("请插入第%d个学生的信息：\n", n);

printf("学号：");

gets(num);

printf("姓名：");

gets(name);

printf("性别：");

gets(sex);

printf("成绩：");

scanf("%d", &score);

strcpy(pnew->num, num);//学号

strcpy(pnew->name, name);//姓名

strcpy(pnew->sex, sex);//性别

pnew->score = score;//成绩

//在tmp和temp之间插入一个新节点

tmp->next = pnew;

pnew->next = temp;

getchar();//Empty the buffer

printf("插入成功\n");

}

else if (n == m)

{

for (i = 0; i < n; i++)

{

tmp = tmp->next;

}

printf("请插入第%d个学生的信息：\n", n);

printf("学号：");

gets(num);

printf("姓名：");

gets(name);

printf("性别：");

gets(sex);

printf("成绩：");

scanf("%d", &score);

strcpy(pnew->num, num);//学号

strcpy(pnew->name, name);//姓名

strcpy(pnew->sex, sex);//性别

pnew->score = score;//成绩

pnew->next = NULL;

tmp->next = pnew;

getchar();//Empty the buffer

printf("插入成功\n");

}

else

printf("请输入不超过%d的数\n", m);

}

/\*删除信息\*/

void delete(stu\* phead)

{

int i = 0;

char num[10];//学号

printf("请输入要删除的学生学号：");

gets(num);

stu\* tmp = phead;

stu\* temp = phead;

while (tmp)

{

//遍历链表查找

if (strcmp(tmp->num, num) != 0)

{

temp = tmp;

tmp = tmp->next;

if (tmp == NULL)

{

printf("删除失败！\n");

return;

}

}

else

break;

}

temp->next = tmp->next;

free(tmp);

printf("删除成功！\n");

}

/\*查询信息\*/

void search(stu\* phead)

{

int i = 1;//用于标记是否查询到学生信息

char num[10];//学号

printf("请输入要查询学生的学号：");

gets(num);

stu\* tmp = phead->next;

while (tmp)

{

if (strcmp(tmp->num, num) == 0)

{

i = 0;

printf("该学生的姓名：");

printf("%s\n", tmp->name);

printf("该学生的性别：");

printf("%s\n", tmp->sex);

printf("该学生的成绩：");

printf("%d\n", tmp->score);

}

tmp = tmp->next;

}

if (i)

printf("未查询到该学生！\n");

}

