#include <stdio.h>

#include <malloc.h>

#include <stdlib.h>

typedef int datatype;

typedef struct node

{ datatype data;

struct node \*next;

}slnode;

int l; /\*保存查找到的结点位置\*/

int initiate (slnode \*\*h );

int append(slnode \*h);

int insert(slnode \*h, int n);

int deletet(slnode \*h, int n);

slnode \* search(slnode \*h, datatype x);

void print(slnode \*h);

void main()

{ slnode \*h,\*s;

int sel,n,x; /\* sel 选择输入， n 插入或删除结点的个数，x 要查找的结点数据\*/

initiate ( &h );

printf("请输入你的选择: 1---initiate 2---append 3---insert 4---delete 5--- search 6---print 7---exit\nyour choice =");

scanf("%d",&sel);

while(sel!=7)

{ if(sel==1)

{ initiate ( &h ); }

else if(sel==2)

{ append(h);

print(h); }

else if(sel==3)

{ printf("请输入待插入的结点个数:");

scanf("%d",&n);

insert(h,n);

print(h); }

else if(sel==4)

{

deletet(h,x);

print(h); }

else if(sel==5)

{ printf("请输入要查找的结点数据:");

scanf("%d",&x);

s=search(h,x);

if(s!=NULL)

printf("查找成功，结点位置为：%d\n",l); }

else if(sel==6)

{ print(h); }

else

printf("你的选择是错误的!\n ");

printf("请输入你的选择: 1---initiate 2---append 3---insert 4---delete 5--- search 6---print 7---exit \nyour choice =");

scanf("%d",&sel); } }

/\*初始化\*/

int initiate (slnode \*\*h)

{ \*h=( slnode \* )malloc(sizeof(slnode));

if (\*h==NULL)

{ printf("初始化失败！\n");

return 0; }

(\*h)->next=NULL;

printf("完成初始化 !\n");

return 1; }

/\*建立单链表\*/

int append (slnode \*h)

{ slnode \*p,\*q;

int n,i; /\* n 单链表中最初的结点数\*/

datatype a[10]; /\* a 单链表中最初的数据\*/

if(h==NULL)

{ printf("单链表未初始化！\n");

return 0; }

printf("请输入单链表最初的结点数n=");

scanf("%d",&n);

printf("请输入各个结点数据域的值:");

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

{ scanf("%d",&a[i]); }

p=h;

for(i=0;i<=n-1;i++)

{ if ((q=( slnode \* )malloc(sizeof(slnode)))==NULL)

{ printf("新结点空间申请失败！\n");

return 0; }

q->data=a[i];

q->next=NULL;

p->next=q;

p=q; }

return 1; }

/\*插入\*/

int insert (slnode \*h, int n)

{ slnode \*p, \*q;

int i, j; /\* i 插入结点的位置\*/

datatype x; /\* x 插入结点的数据\*/

p=h;

j=0;

while(n)

{ printf("请输入插入结点的位置及结点数据：");

scanf("%d%d",&i,&x);

while((p->next!=NULL)&&(j<i-1))

{ p=p->next;

j++; }

if(j!=i-1)

{ printf("\n插入位置不合理！\n"); }

else

{ if ((q=( slnode \* )malloc(sizeof(slnode)))==NULL)

{ printf("新结点空间申请失败！\n");

return 0; }

q->data=x;

q->next=p->next;

p->next=q;

n--; } }

printf("插入完成!\n");

return 1; }

/\*删除\*/

int deletet (slnode \*h,datatype x)

{ slnode \*p,\*q;

p=h;

while (p->next)

{

if(p->next->data%2!=0)

{ q=p->next;

p->next=q->next;

free(q);

/\*p=p->next;\*/

}

else

p=p->next;

}

printf("删除完成!\n");

return 1;

}

/\*查找\*/

slnode \* search(slnode \*h, datatype x)

{ slnode \*p;

p=h;

l=0;

while(p->next!=NULL)

{ l++;

if(p->next->data==x)

return p;

p=p->next; }

printf("没有要删除的结点数据！\n");

return NULL; }

/\*显示输出\*/

void print(slnode \*h)

{ slnode \*p;

int i;

p=h->next;

i=0;

printf("\n当前的单链表数据内容为:\n");

while(p!=NULL)

{ i++;

printf("第%d个数据： %d \n",i,p->data);

p=p->next; }

printf("\n线性表的表长为%d\n",i); }