**填空题**

1. 以下printlist函数输出链表所有结点的信息。

struct node

{ /\*链表结点定义\*/

int num; /\*学号\*/

char name[20]; /\*姓名\*/

struct node \*next; /\*指向下一结点的指针\*/

};

void printlist(struct node \*head)

{ /\*输出链表的内容\*/

struct node \*p;

p = head;

while ( )

{

printf( "%d %s\n", p->num, p->name );

p = ;

}

}

2 . 以下程序中insert1函数**将新结点插入到链表的表头处**，printlist函数输出链表的内容，主函数main用于测试两个函数。

#include <stdio.h>

#include <string.h>

#include <stdlib.h>

struct node

{ /\*链表结点定义\*/

int num; /\*学号\*/

char name[20]; /\*姓名\*/

struct node \*next; /\*指向下一结点的指针\*/

};

struct node \*insert1(struct node \*head, int num, char \*name)

{ /\*将结点插入到链表的表头，并返回新链表的表头\*/

struct node \*p;

p = ;

p->num = num;

strcpy( p->name, name );

p->next = head;

return ;

}

void printlist(struct node \*head)

{ /\*输出链表的内容\*/

struct node \*p;

for ( p=head; p!=NULL; p=p->next )

printf( "%d\t%s\n", p->num, );

}

main()

{ /\*先定义一个空链表，接着插入三个结点，然后输出链表\*/

struct node \*head = NULL;

head = insert1( head, 111, "wang" );

head = insert1( head, 222, "chen" );

head = insert1( head, 333, "gong" );

printlist( head );

/\*链表的结点删除部分略\*/

}

程序运行结果为：

333 gong

222 chen

111 wang

**3.** 以下结点node定义了一个学生的信息，函数delsub从链表head1中删去所有出现在head2链表中的学生，并返回删除后链表的头指针。

struct node /\*学生结点定义\*/

{

int num; /\*学号\*/

struct node \*next;

};

struct node \*delsub(struct node \*head1,struct node \*head2)

{

struct node \*p,\*q,\*r;

for (r=head2;r!=NULL;r=r->next) /\*遍历head2所有结点\*/

{

if (head1==NULL)

break;

p=head1;

if (p->num==r->num)

{

head1= \_\_\_\_;

free(p);

}

else for ( ; \_\_\_\_ ; p=p->next)

{

q=p->next;

if (q->num==r->num)

{

p->next= \_\_\_\_\_\_\_\_\_;

free(q);

break;

}

}

}

return head1;

}

4. 以下结点node定义了一个学生的信息，**函数insert将一个新结点插入到链表head的末尾**并返回新链表的头结点指针。函数list输出链表中所有结点的学号信息。

#include <stdio.h>

#include <stdlib.h>

struct node

{

int num;

struct node \*next;

};

struct node \*insert(struct node \*head,int num)

{

struct node \*p,\*q;

p = ;

p->num = num;

p->next = NULL;

if (head==NULL)

return p;

for (q=head; ;q=q->next)

;

q->next = p;

return ;

}

void list(struct node \*head)

{

struct node \*p;

for (p=head;p!=NULL;p=p->next)

printf("%d\t",p->num);

}

main()

{

struct node \*head=NULL;

int i;

for (i=10;i<=15;i++)

head=insert(head,i);

list(head);

}

/\*运行后将输出: 10 11 12 13 14 15\*/