C++实现链表

#include<stdio.h>

typedef struct \_NODE\_

{

char a;

\_NODE\_ \* next;

}Node,\*pNode;

pNode Head = NULL;

pNode Tail = NULL;

void CreateLinkNode(char a);

void main()

{

char temp = 'a';

for(int i=0;i<5;i++)

{

CreateLinkNode(temp+i);

}

int Count=0;

pNode pNodeTemp = Head;

while(pNodeTemp!=NULL)

{

Count++;

printf("%c ",pNodeTemp->a);

pNodeTemp = pNodeTemp->next;

}

printf("链表长度为%d\n",Count);

int m = 2;

pNodeTemp = Head;

while(m)

{

pNodeTemp = pNodeTemp->next;

m--;

}

printf("第三个元素为 %c \n",pNodeTemp->a);

pNodeTemp = Head;

int Pos=0;

while(pNodeTemp)

{

Pos++;

if(pNodeTemp->a == 'a')

{

printf("a的位置为%d\n",Pos);

break;

}

pNodeTemp = pNodeTemp->next;

}

pNodeTemp = Head;

for(int k=0;k<2;k++)

{

pNodeTemp = pNodeTemp->next;

}

pNode pNodeNew = new Node;

if(pNodeNew==NULL)

{

return ;

}

pNodeNew->a = 'f';

pNodeNew->next = NULL;

pNodeNew->next = pNodeTemp->next;

pNodeTemp->next = pNodeNew;

Count++;

pNodeTemp = Head;

while(pNodeTemp!=NULL)

{

printf("%c ",pNodeTemp->a);

pNodeTemp = pNodeTemp->next;

}

pNodeTemp = Head->next;

pNodeTemp->next = pNodeTemp->next->next;

pNodeTemp = Head;

printf("\n");

while(pNodeTemp!=NULL)

{

printf("%c ",pNodeTemp->a);

pNodeTemp = pNodeTemp->next;

}

while(Head!=Tail)

{

pNodeTemp = Head;

Head = Head->next;

delete pNodeTemp;

}

delete Head;

}

void CreateLinkNode(char a)

{

pNode pNodeTemp = new Node;

if(pNodeTemp==NULL)

{

return ;

}

pNodeTemp->a = a;

pNodeTemp->next = NULL;

if(Head==NULL)

{

Head=Tail = pNodeTemp;

}

else

{

Tail->next = pNodeTemp;

Tail = pNodeTemp;

}

}