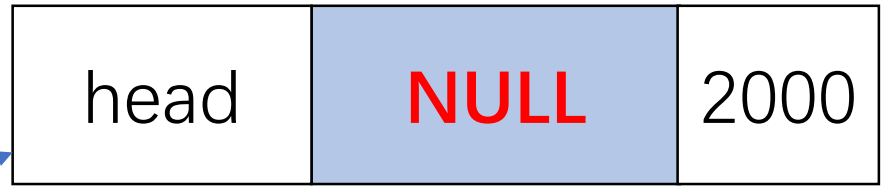
# 1650254 尤尧寅 计算机一班
# 链表指针思考题

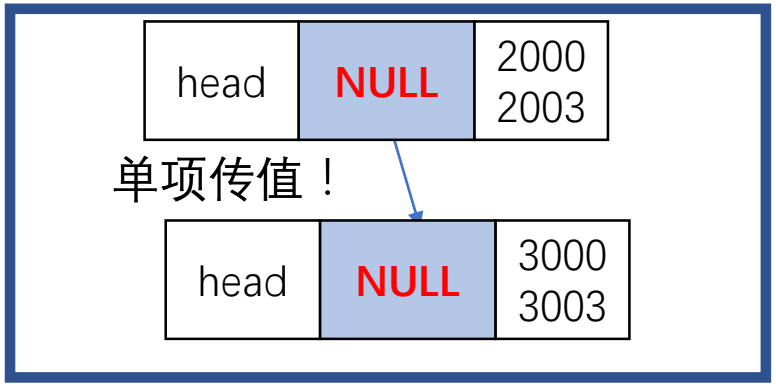
# 源程序过程

- （设初位地址为2000）

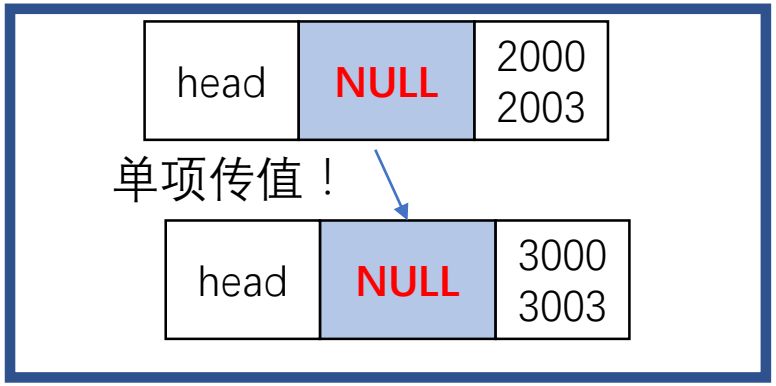


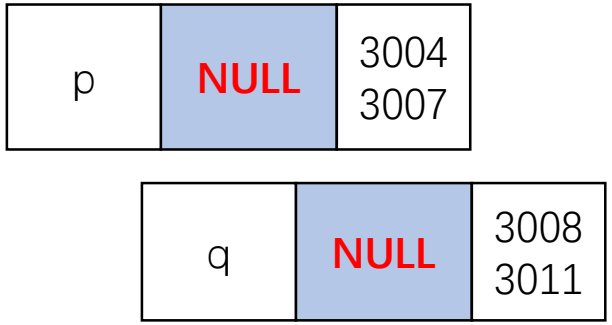

```
int main()
{
student *head = NULL;

if (linklist_create(head) == OK) {。。。。。

    }
}
```

```
int linklist_create(student *head)
{
student *p = NULL, *q = NULL;
int i;

for (i = 0; i < 5; i++) {
if (i > 0)
q = p;
p = new(nothrow) student; //思考：为什么不能用
malloc
if (p == NULL)
return ERROR;
if (i == 0)
head = p; //head指向第1个结点
else
q->next = p;
cout << "请输入第" << i + 1 << "个人的基本信息" <<
endl;
cin >> p->name >> p->num >> p->sex; //键盘输入
基本信息
p->next = NULL;
}
return OK;
}
```

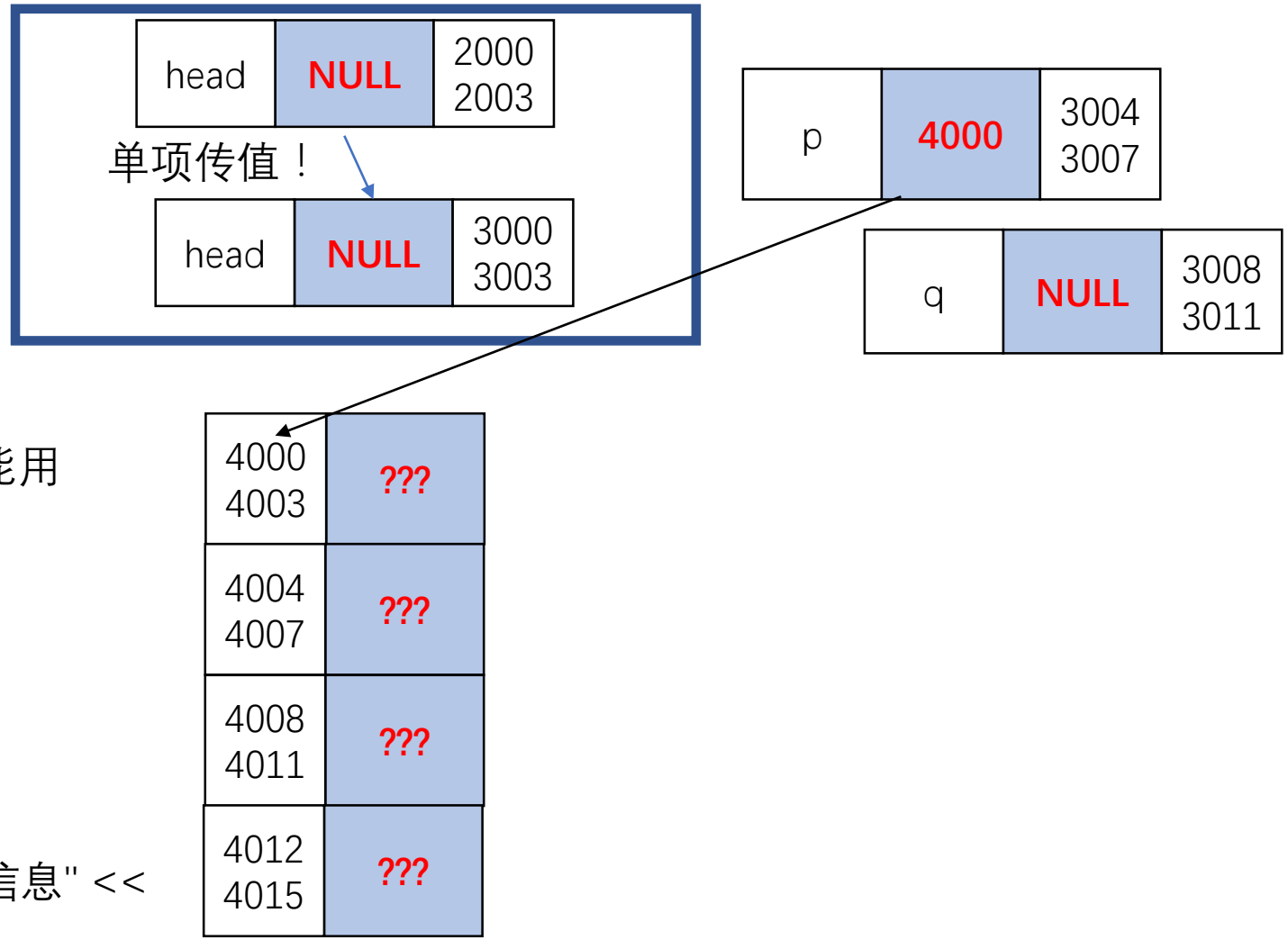


```
int linklist_create(student *head)
{
student *p = NULL, *q = NULL;
int i;

for (i = 0; i < 5; i++) {
if (i > 0)
q = p;
p = new(nothrow) student; //思考：为什么不能用
malloc
if (p == NULL)
return ERROR;
if (i == 0)
head = p; //head指向第1个结点
else
q->next = p;
cout << "请输入第" << i + 1 << "个人的基本信息" <<
endl;
cin >> p->name >> p->num >> p->sex; //键盘输入
基本信息
p->next = NULL;
}
return OK;
}
```

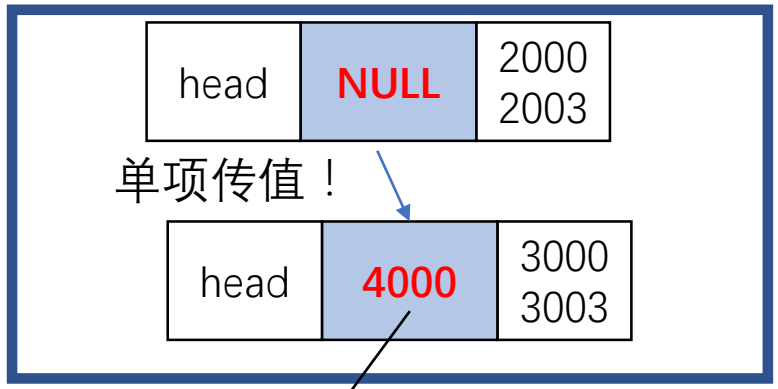


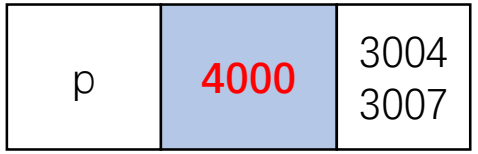



```
int linklist_create(student *head)
{
student *p = NULL, *q = NULL;
int i;

for (i = 0; i < 5; i++) {
if (i > 0)
q = p;
p = new(nothrow) student; //思考：为什么不能用
malloc
if (p == NULL)
return ERROR;
if (i == 0)
head = p; //head指向第1个结点
else
q->next = p;
cout << "请输入第" << i + 1 << "个人的基本信息" <<
endl;
cin >> p->name >> p->num >> p->sex; //键盘输入
基本信息
p->next = NULL;
}
return OK;
}
```

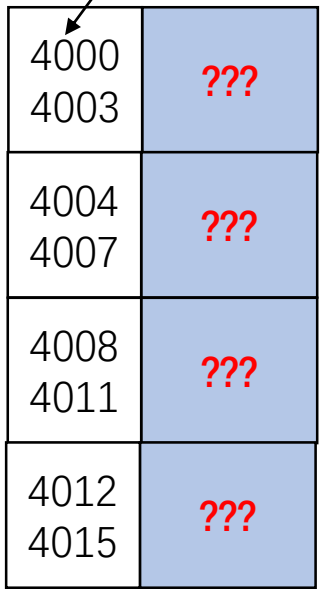


```
int linklist_create(student *head)
{
student *p = NULL, *q = NULL;
int i;

for (i = 0; i < 5; i++) {
if (i > 0)
q = p;
p = new(nothrow) student; //思考：为什么不能用
malloc
if (p == NULL)
return ERROR;
if (i == 0)
head = p; //head指向第1个结点
else
q->next = p;
cout << "请输入第" << i + 1 << "个人的基本信息" <<
endl;
cin >> p->name >> p->num >> p->sex; //键盘输入
基本信息
p->next = NULL;
}
return OK;
}
```

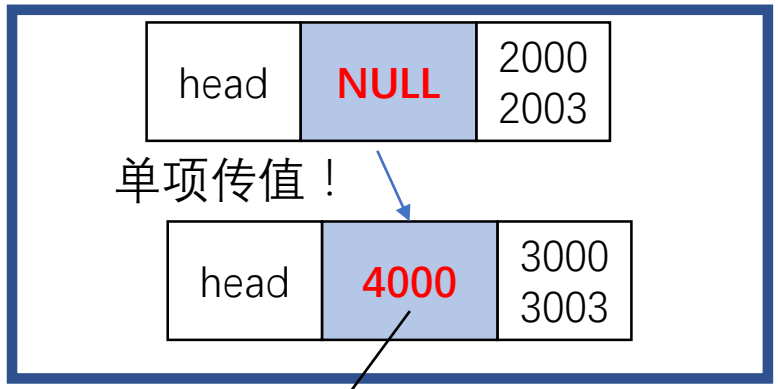


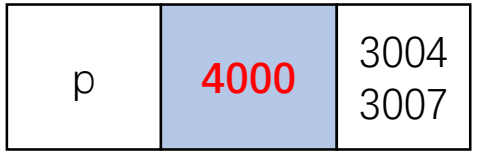





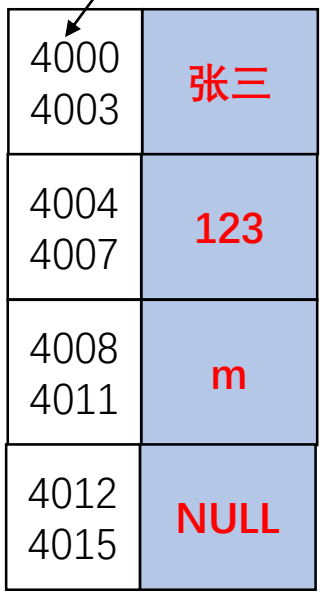


```
int linklist_create(student *head)
{
student *p = NULL, *q = NULL;
int i;

for (i = 0; i < 5; i++) {
if (i > 0)
q = p;
p = new(nothrow) student; //思考：为什么不能用
malloc
if (p == NULL)
return ERROR;
if (i == 0)
head = p; //head指向第1个结点
else
q->next = p;
cout << "请输入第" << i + 1 << "个人的基本信息" <<
endl;
cin >> p->name >> p->num >> p->sex; //键盘输入
基本信息
p->next = NULL;
}
return OK;
}
```

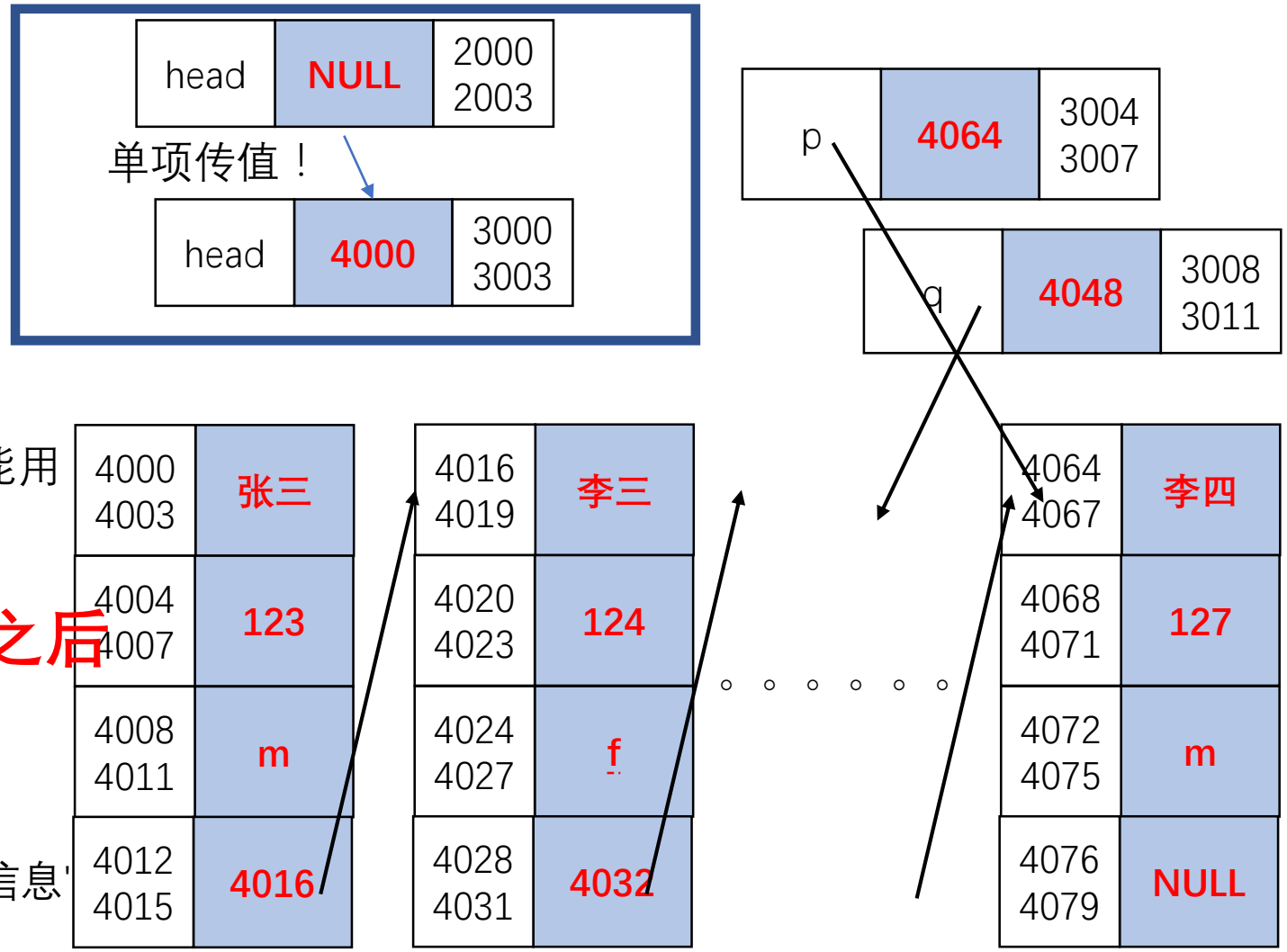




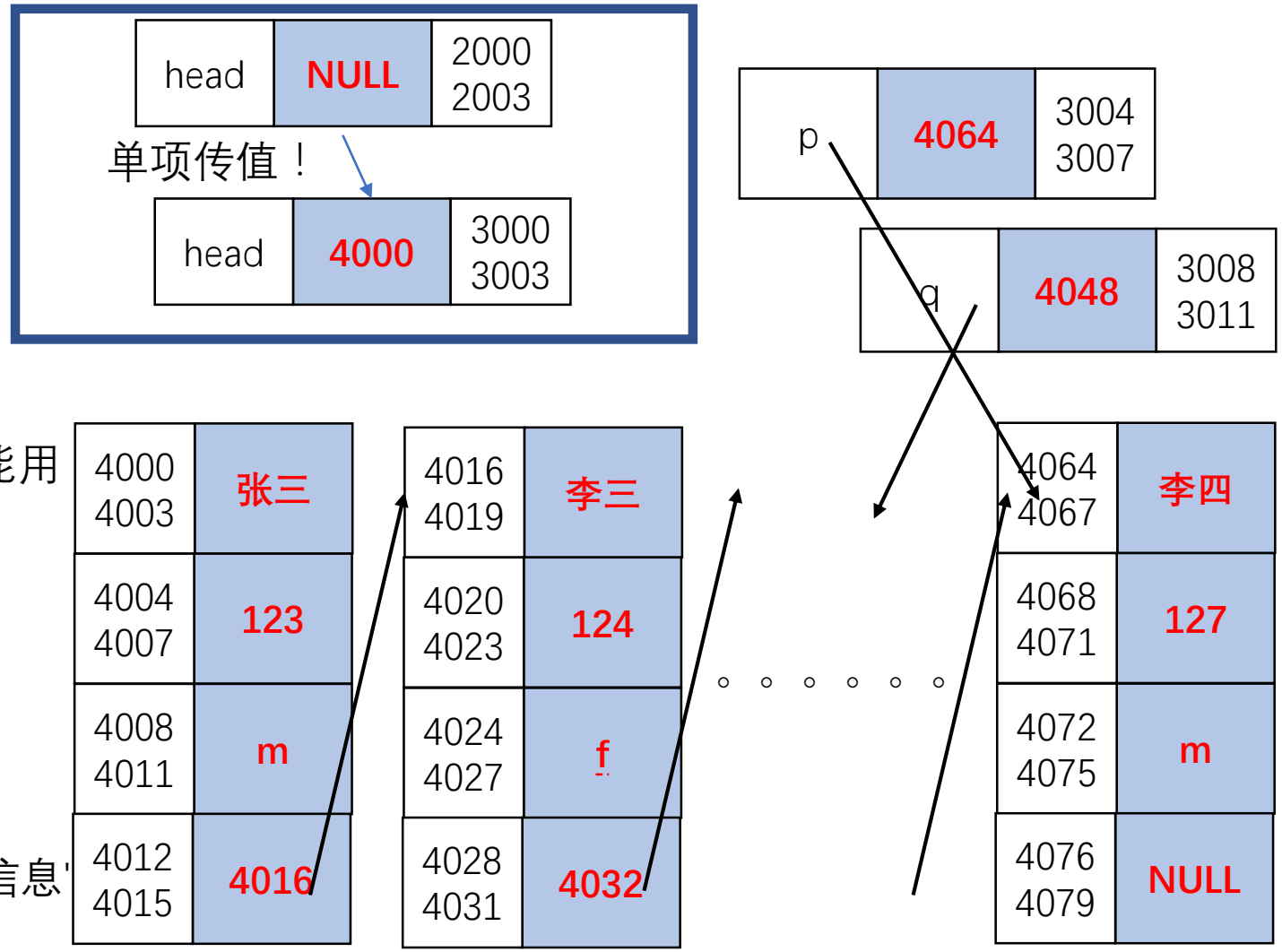


```
int linklist_create(student *head)
{
student *p = NULL, *q = NULL;
int i;

for (i = 0; i < 5; i++) {
if (i > 0)
q = p;
p = new(nothrow) student; //思考：为什么不能用
malloc
if (p == NULL)
return ERROR;
if (i == 0)
head = p; //head指向第1个结点
else
q->next = p;
cout << "请输入第" << i + 1 << "个人的基本信息"
endl;
cin >> p->name >> p->num >> p->sex; //键盘输入
基本信息
p->next = NULL;
}
return OK;
}
```



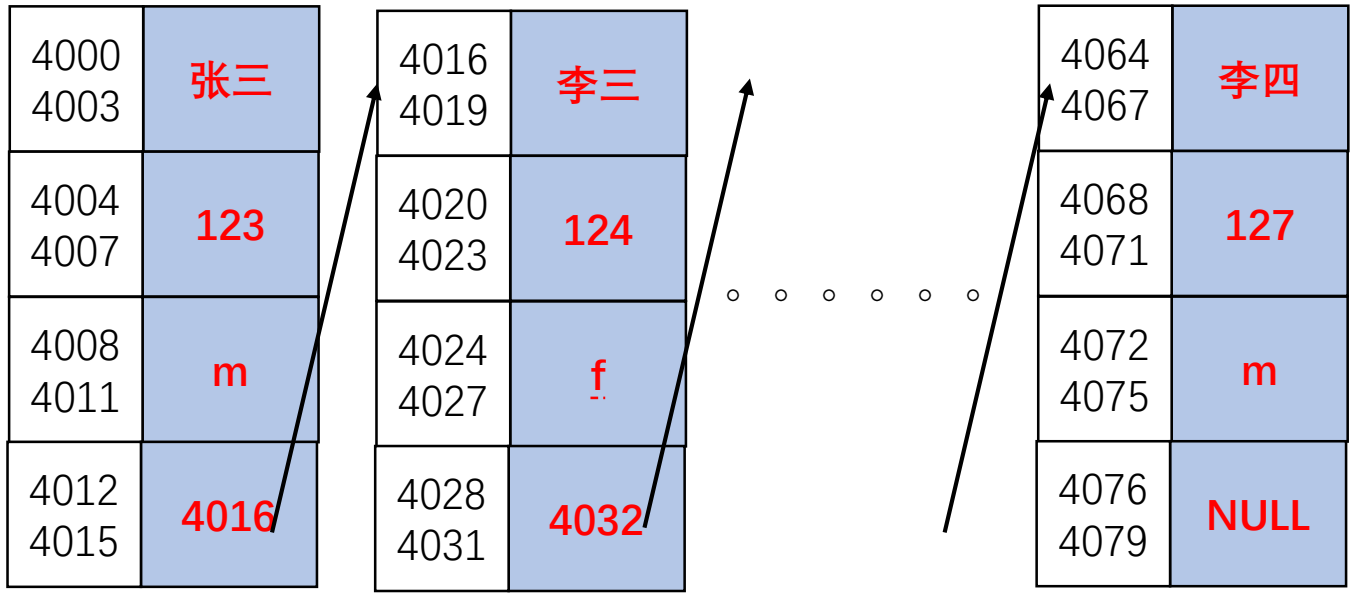

**重复五次之后**

```
int linklist_create(student *head)
{
student *p = NULL, *q = NULL;
int i;

for (i = 0; i < 5; i++) {
if (i > 0)
q = p;
p = new(nothrow) student; //思考：为什么不能用
malloc
if (p == NULL)
return ERROR;
if (i == 0)
head = p; //head指向第1个结点
else
q->next = p;
cout << "请输入第" << i + 1 << "个人的基本信息"
endl;
cin >> p->name >> p->num >> p->sex; //键盘输入
基本信息
p->next = NULL;
}
return OK;
}
```

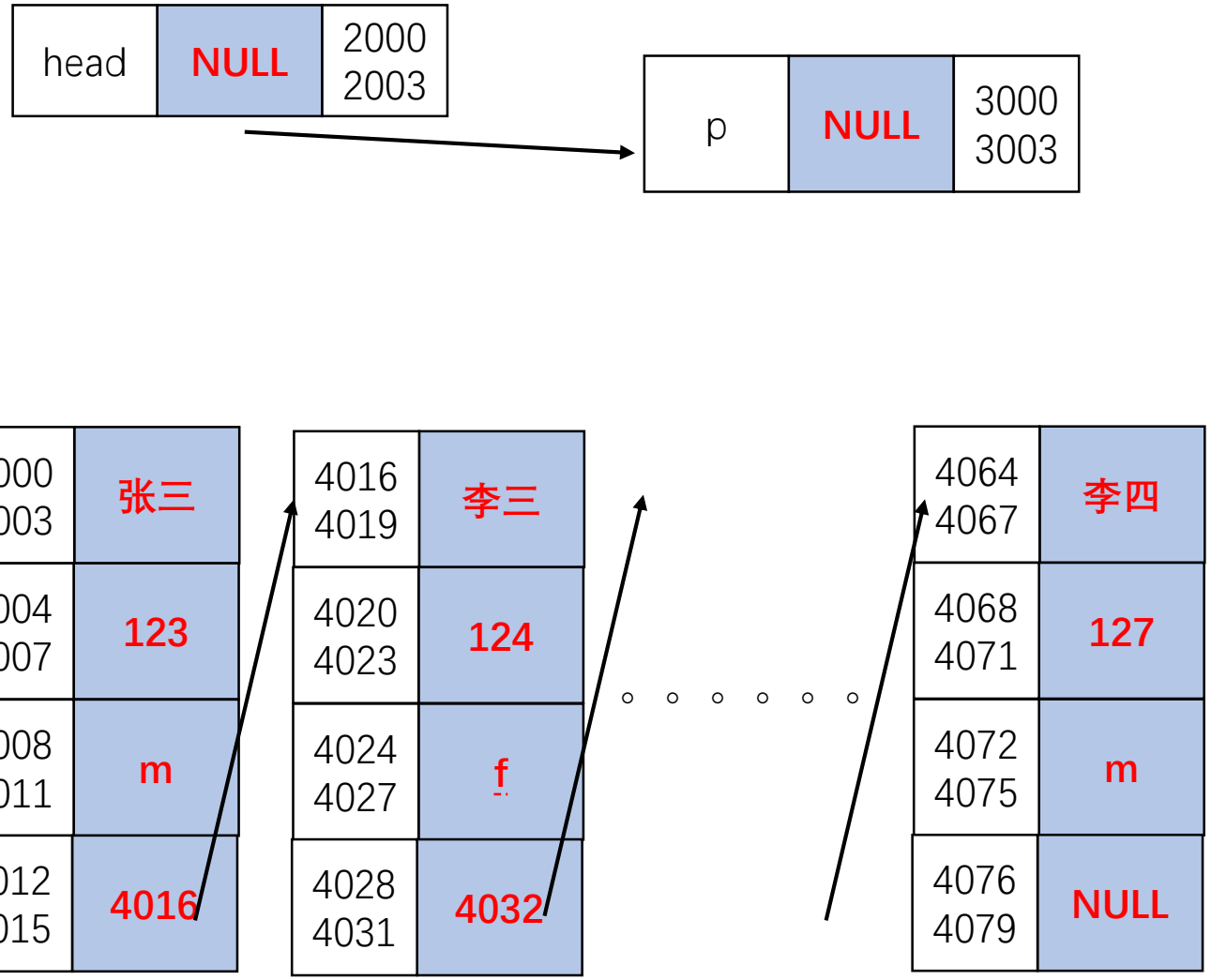


```
int main()
{
student *head = NULL;

if (linklist_create(head) == OK) {
linklist_traverse(head);
linklist_destroy(head);
}
else
cout << "LinkList Create failed." << endl;

return 0;
}
```

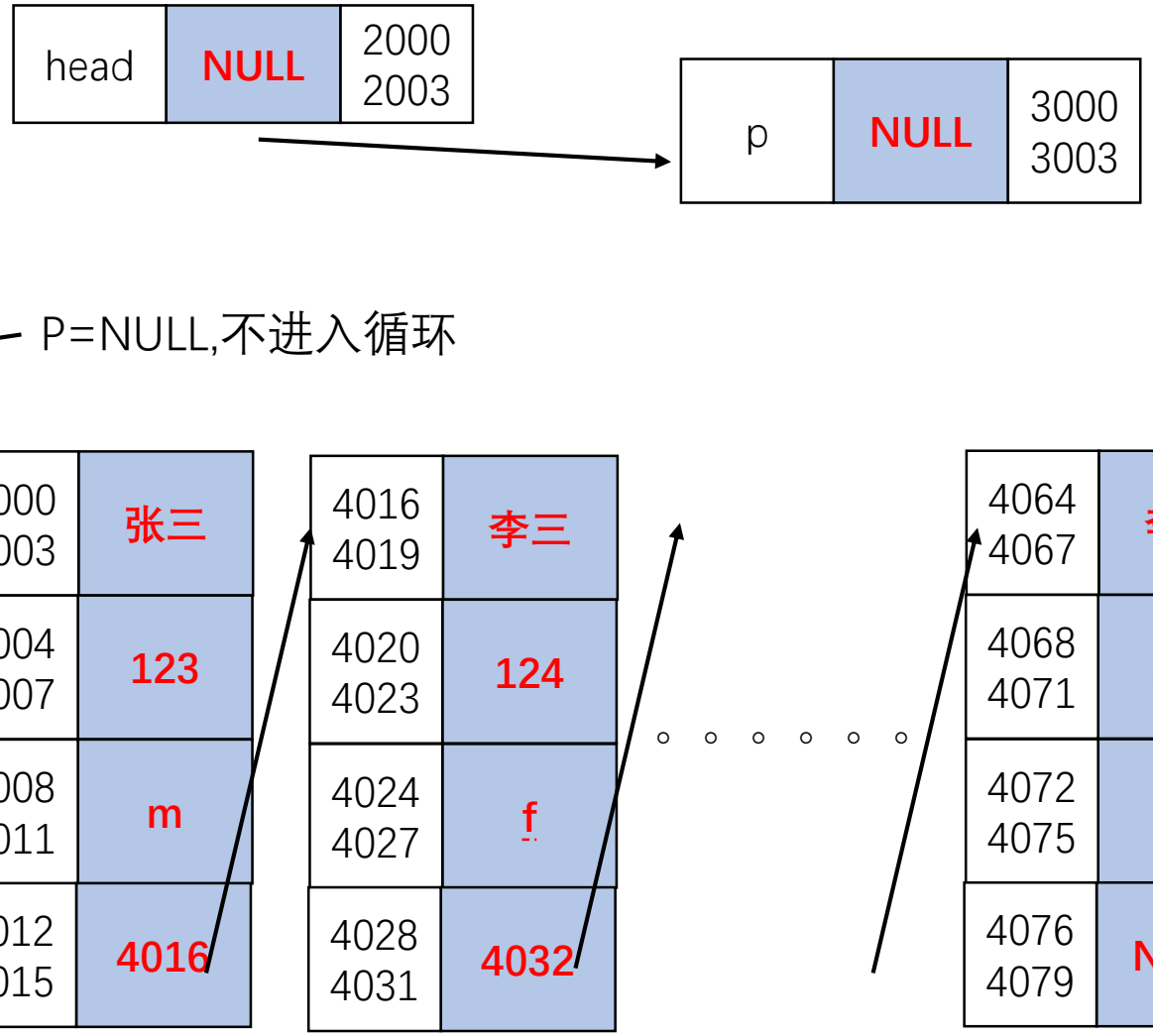

Head的值没有发生变化
局部变量空间被收回
动态空间保留



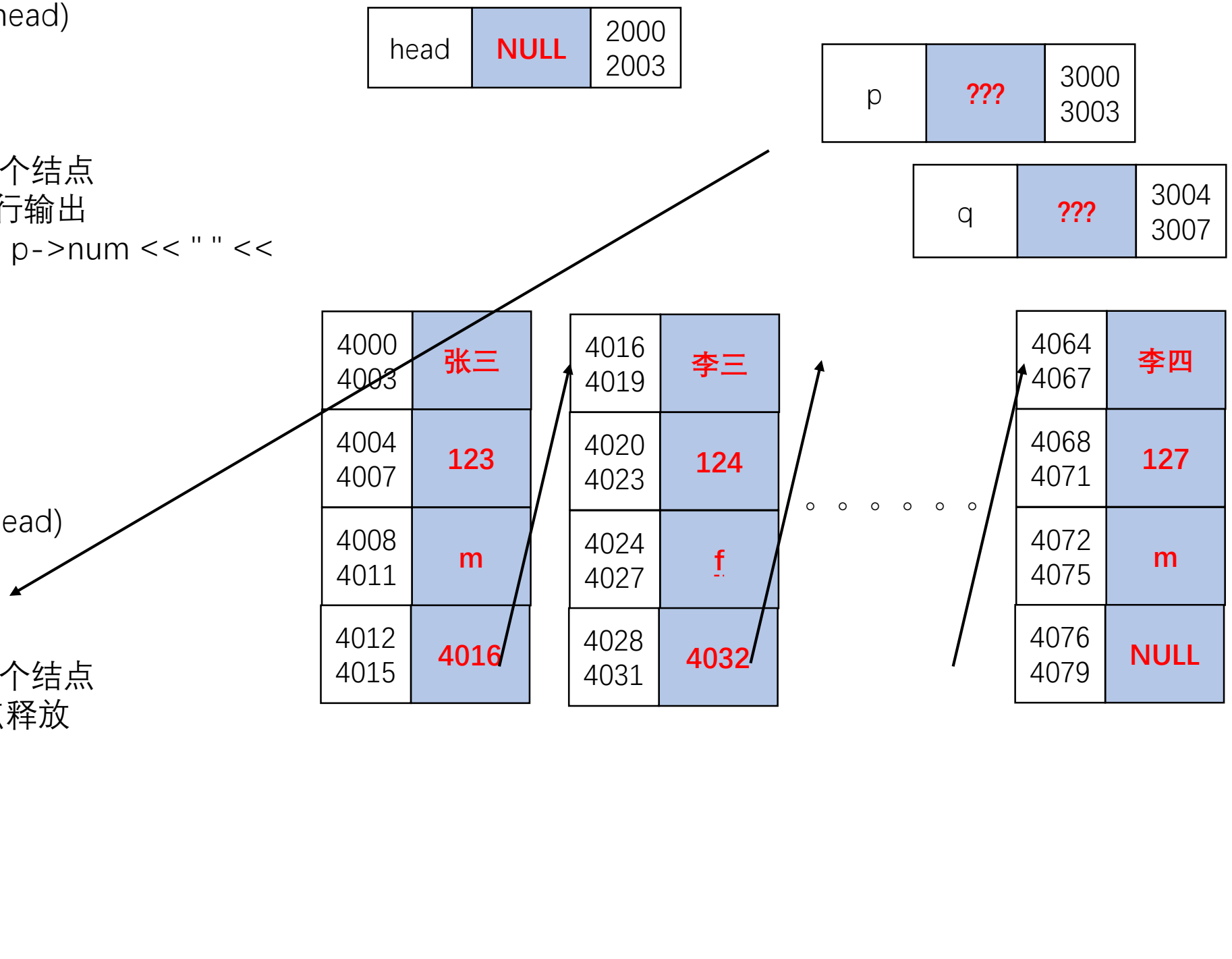
```
int linklist_traverse(student *head)
{
student *p;

p = head; //p复位，指向第1个结点
while (p != NULL) { //循环进行输出
cout << p->name << " " << p->num << " " <<
p->sex << endl;
p = p->next;
}
return OK;
}

int linklist_destroy(student *head)
{
student *p, *q;

p = head; //p复位，指向第1个结点
while (p) {  //循环进行各结点释放
q = p->next;
delete p;
p = q;
}
return OK;
}
```



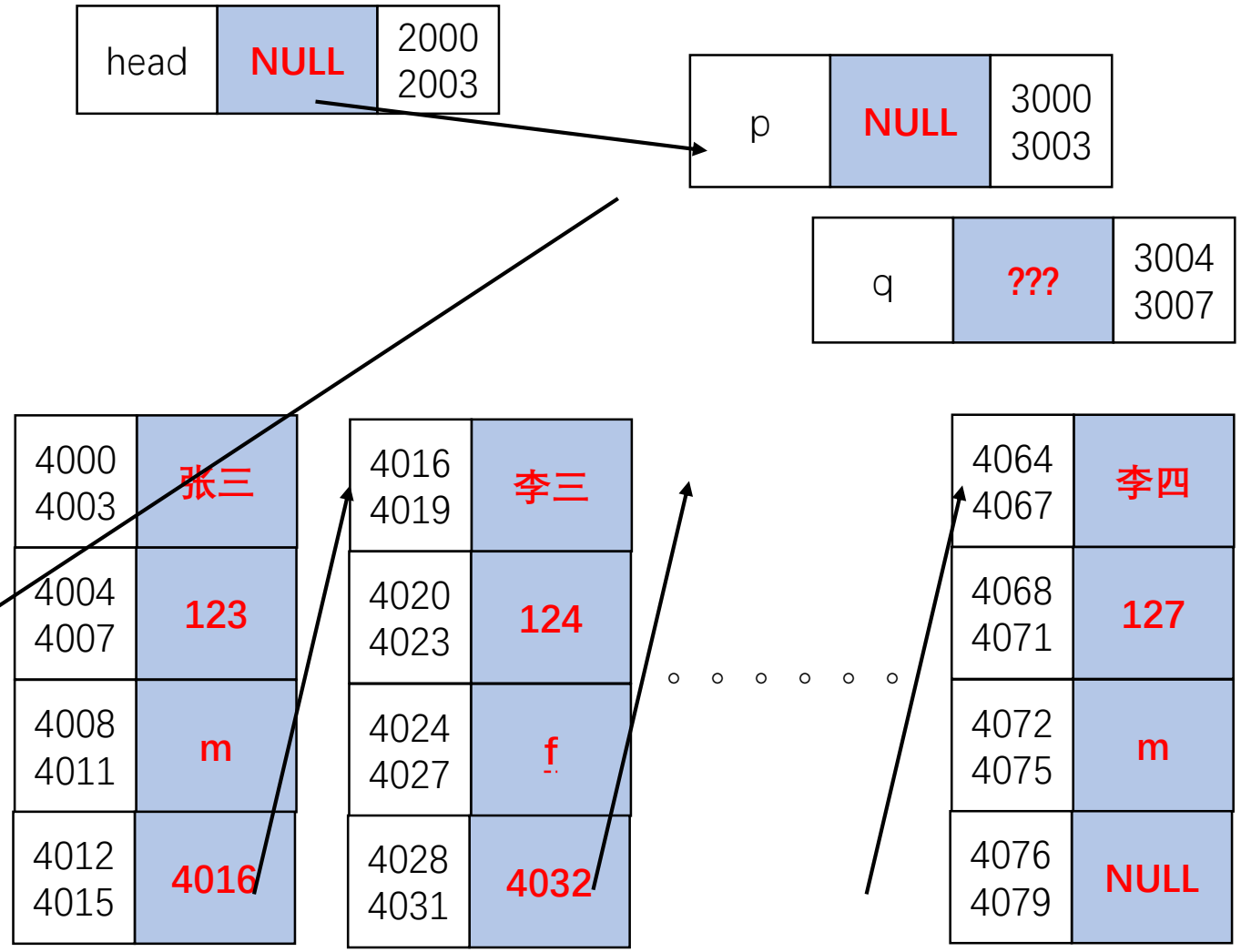
```
int linklist_traverse(student *head)
{
student *p;

p = head; //p复位，指向第1个结点
while (p != NULL) { //循环进行输出
cout << p->name << " " << p->num << " " <<
p->sex << endl;
p = p->next;
}
return OK;
}

int linklist_destroy(student *head)
{
student *p, *q;

p = head; //p复位，指向第1个结点
while (p) {  //循环进行各结点释放
q = p->next;
delete p;
p = q;
}
return OK;
}
```



```
int linklist_traverse(student *head)
{
student *p;

p = head; //p复位，指向第1个结点
while (p != NULL) { //循环进行输出
cout << p->name << " " << p->num << " " <<
p->sex << endl;
p = p->next;
}
return OK;
}

int linklist_destroy(student *head)
{
student *p, *q;

p = head; //p复位，指向第1个结点
while (p) {  //循环进行各结点释放
q = p->next;
delete p;
p = q;
}
return OK;
}
```



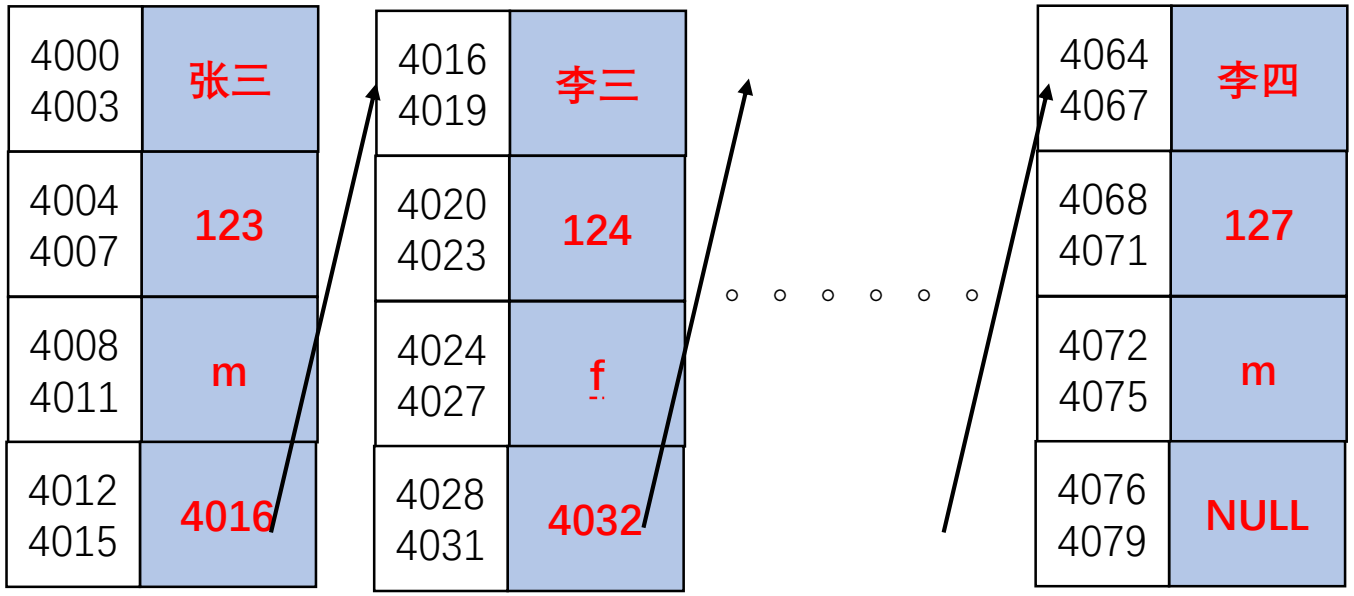
```
int linklist_traverse(student *head)
{
student *p;

p = head; //p复位，指向第1个结点
while (p != NULL) { //循环进行输出
cout << p->name << " " << p->num << " " <<
p->sex << endl;
p = p->next;
}
return OK;
}

int linklist_destroy(student *head)
{
student *p, *q;

p = head; //p复位，指向第1个结点
while (p) {  //循环进行各结点释放
q = p->next;
delete p;
p = q;
}
return OK;
}
```



```
int linklist_traverse(student *head)
{
student *p;

p = head; //p复位，指向第1个结点
while (p != NULL) { //循环进行输出
cout << p->name << " " << p->num << " " <<
p->sex << endl;
p = p->next;
}
return OK;
}

int linklist_destroy(student *head)
{
student *p, *q;

p = head; //p复位，指向第1个结点
while (p) {  //循环进行各结点释放
q = p->next;
delete p;
p = q;
}
return OK;
}
```

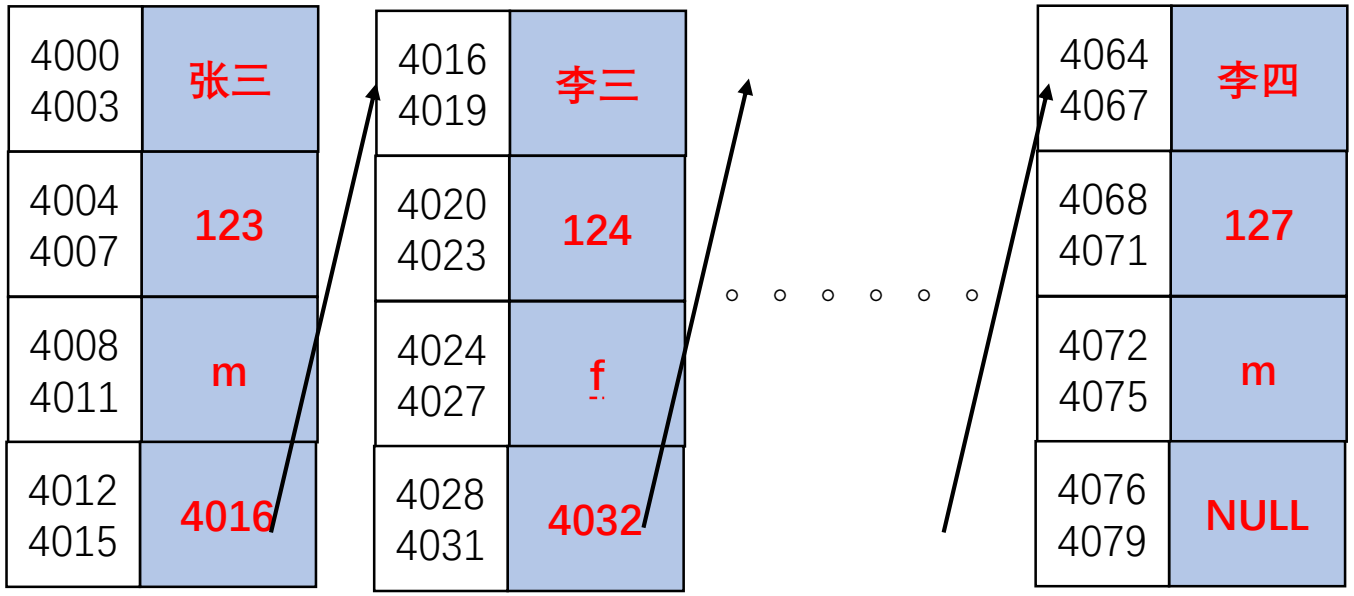



不进入循环




链表没有销毁成功


```
int main()
{
student *head = NULL;

if (linklist_create(head) == OK) {
linklist_traverse(head);
linklist_destroy(head);
}
else
cout << "LinkList Create failed." << endl;

return 0;
}
```



# 1.链表的建立是否成功

- 成功，但因为head没有被储存下来，造成内存丢失

# 2.为什么遍历没有成功

- Head单向传值，主函数的head依然是NULL，在函数里直接没有进入循环

# 3.链表的销毁是否成功

- 不成功 传入的头指针不对

# 4.程序是否有内存丢失情况

- 有，在创建链表之后，即int linklist_create(student *head)之后，头指针丢失，无法释放内存

# 修改后代码

```
#include <iostream>
#include <string> //C++ string类 需要用到
using namespace std;

struct student {
string name;
int num;
char sex;
struct student *next;//指向结构体自身的指针(下个结点)
};

#define OK1
#define ERROR0

int linklist_create(student **head);//修改
int linklist_traverse(student *head);
int linklist_destroy(student *head);

int linklist_create(student **head)//修改
{
student *p = NULL, *q = NULL;
int i;

for (i = 0; i < 5; i++) {
if (i > 0)
q = p;
p = new(nothrow) student; //思考：为什么不能用malloc  答：使用了string类
if (p == NULL)
return ERROR;
if (i == 0)
*head = p; //head指向第1个结点)//修改
else
q->next = p;
cout << "请输入第" << i + 1 << "个人的基本信息" << endl;
cin >> p->name >> p->num >> p->sex; //键盘输入基本信息
p->next = NULL;
}
return OK;
}

int linklist_traverse(student *head)
{
student *p;

p = head; //p复位，指向第1个结点
while (p != NULL) { //循环进行输出
cout << p->name << " " << p->num << " " << p->sex <<
endl;
p = p->next;
}
return OK;
}

int linklist_destroy(student *head)
{
student *p, *q;

p = head; //p复位，指向第1个结点
while (p) {  //循环进行各结点释放
q = p->next;
delete p;
p = q;
}
return OK;
}

int main()
{
student *head = NULL;

if (linklist_create(&head) == OK) {//修改
linklist_traverse(head);
linklist_destroy(head);
}
else
cout << "LinkList Create failed." << endl;

return 0;
}
```