

Ôn tập Cấu trúc dữ liệu và giải thuật

SV: Nguyễn Hữu Sơn

www.facebook.com/nhsonit

*Em xin trân thành cảm ơn thầy **Nguyễn Văn Thắng** đã nhiệt tình chỉ dạy để em hoàn thành tài liệu ôn tập này.*

Lời chúc của thầy đến sinh viên khoa CNTT-K10 :

Chúc các em đạt kết quả cao nhất trong các kỳ thi nói chung và trong môn học CTDL> nói riêng.

Quá trình học tập và rèn luyện nào cũng luôn đồng hành với những khó khăn với những thành công, thất bại, có thể gây nản chí cho bất cứ ai. Vì vậy mỗi chúng ta cần tìm cho mình phương thức học tập chủ động, hợp lý, sáng tạo.

Thầy mong các em luôn phấn đấu không ngừng, đoàn kết, giúp đỡ nhau trong học tập và trong cuộc sống; mỗi người hãy "vì mình" và "vì mọi người" nhiều hơn nữa nhé.

Hãy tự hào mình là một sinh viên ĐH CNTT K10 đã, đang và sẽ góp phần làm rạng rỡ truyền thống của Khoa Công nghệ thông tin nói riêng và của Trường Đại học Công nghiệp Hà Nội nói chung.

Phần 1: Đề Quy

Bài 1:

Xét định nghĩa đệ quy

$$Acker(m, n) = \begin{cases} n + 1 & \text{nếu } m = 0 \\ Acker(m - 1, 1) & \text{nếu } n = 0 \\ Acker(m - 1, Acker(m, n - 1)) & \text{trường hợp khác} \end{cases}$$

a. Hãy **sác** định $Acker(1, 2)$

(sai chính tả nhé, xác định, không phải SÁC nhé)

b. Viết hàm đệ Quy thực hiện tính giá trị của hàm này

Bài Giải:

$$Acker(1, 2) = Acker(1-1, Acker(1, 2-1)) = Acker(0, Acker(1, 1)) \quad (1)$$

$$Acker(1, 1) = Acker(0, Acker(1, 0)) \quad (2)$$

$$AAcker(1, 0) = Acker(0, 1) \text{ vì } n=0$$

$$Acker(0, 1) = 1+1=2 \text{ vì } m=0 \text{ thay vào (2) Ta được } Acker(0, 2)$$

$$Acker(0, 2) = 2+1=3 \text{ vì } m=0 \text{ thay vào (1) Ta được } Acker(0, 3)$$

$$Acker(0, 3) = 3+1=4 \text{ vì } m=0$$

$$\text{Vậy } Acker(1, 2) = 4;$$

```
int Acker(int m, int n)
```

```
{
```

```
    if(m==0)
```

```
    {
```

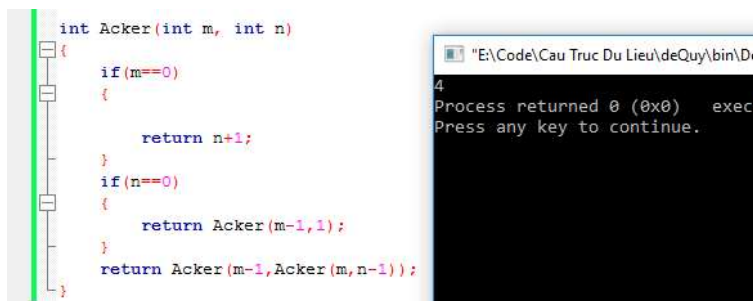
```
        return n+1;
```

```

    }
    if(n==0)
    {
        return Acker(m-1,1);
    }
    return Acker(m-1,Acker(m,n-1));
}

```

Kết quả chạy đoạn mã:



Bài 2: Cho dãy số $A = \{4, 3, -2, -6, -5, 0, 4, \dots\}$

- **Xây dựng định nghĩa đệ quy cho việc tính A_n ($n \geq 0$)**
- **Xây dựng giải thuật đệ quy tính A_n**
- **Tính A_{10}**

Bài giải:

Chúng ta nhận thấy quy luật của dãy số như sau nếu các bạn lấy

$$3 - 4 - 1 = -2$$

$$-2 - 3 - 1 = -6$$

$$-6 - -2 - 1 = -5$$

.....

$4_{A0}, 3_{A1}, -2_{A2}, -6_{A3}, -5_{A4}, 0_{A5}, 4_{A6}.....$

Ta có tổng quát với $n \geq 2$ thì

$$A_n = A_{n-1} - A_{n-2} - 1$$

Còn với $n \geq 0$ thì

$$A(n) = \begin{cases} 4 & \text{nếu } n = 0 \\ 3 & \text{nếu } n = 1 \\ A(n-1) - A(n-2) - 1 & \text{khác'} \end{cases}$$

Code:

```
#include <iostream>
using namespace std;
int A(int n)
{
    if(n==0)
        return 4;
    if(n==1)
        return 3;
    return A(n-1) - A(n-2) -1;
}
```

Hiển nhiên hàm này đúng rồi, nhưng viết thế nó lại không thể hiện được công thức của mình. Nên viết rõ theo nhánh, đừng tiếc từ khoá else:

```
if(n==0)
    return 4;
```

```

else if(n==1)
    return 3;
else return A(n-1) - A(n-2) -1;

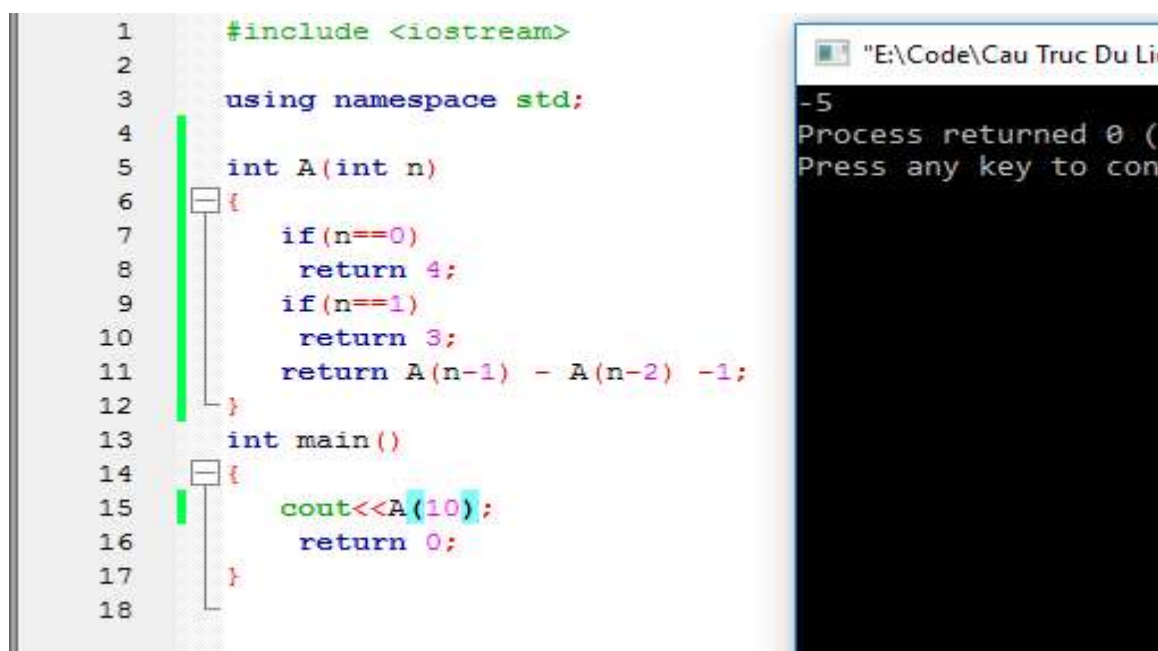
```

```

int main()
{
    cout<<A(10);
    return 0;
}

```

Tính A_{10} các bạn làm tương tự giống bài 1 kết quả $A_{10} = -5$



```

1  #include <iostream>
2
3  using namespace std;
4
5  int A(int n)
6  {
7      if(n==0)
8          return 4;
9      if(n==1)
10         return 3;
11         return A(n-1) - A(n-2) -1;
12     }
13
14     int main()
15     {
16         cout<<A(10);
17         return 0;
18     }

```

Output window: "E:\Code\Cau Truc Du Li
-5
Process returned 0 (Press any key to continue)

Bài 3:

Giải thuật tính ước số chung lớn nhất của hai số nguyên dương p, q ($p > q$) được cho như sau:

Gọi r là số dư trong phép chia p cho q .

Nếu $r = 0$ thì ước số chung lớn nhất là q

Nếu $r \neq 0$ thì gán cho p giá trị q , gán cho q giá trị của r và lặp lại quá trình

- a. Hãy xây dựng định nghĩa đệ quy cho hàm $USCLN(p, q)$.***
- b. Viết một giải thuật đệ quy và một giải thuật lặp thể hiện hàm đó.***
- c. Hãy nêu rõ các đặc điểm của một giải thuật đệ quy được thể hiện trong trường hợp này***
- d. Xử lý trường hợp $p < q$***

Bài giải:

$$r = p \% q;$$

$$USCLN(p, q) = \begin{cases} q & \text{nếu } r = 0 \\ p = q; q = r; USCLN(p, q); & \text{nếu } r \neq 0 \end{cases}$$

Giải thuật đệ quy:

```
int UCLN(int p, int q)
{
    int r = p%q;
    if(r==0)
        return q;
    else
    {
        p = q;
        q = r;
    }
}
```

```
return UCLN(p,q);
```

Không nên gán $p=q$ và $q=r$ như thế này, nó làm mất "hình thức" đệ quy đi, mà nên dùng: `return UCLN(q,r);`

```
}
```

```
}
```

Giải thuật lặp:

```
int _UCLN(int p, int q)
```

```
{
```

```
    int r = p%q;
```

```
    while(r!=0)
```

```
    {
```

```
        p = q;
```

```
        q = r;
```

```
        r = p%q;
```

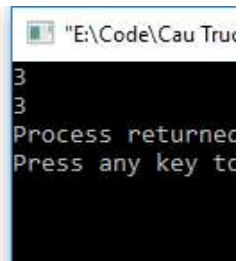
```
    }
```

```
    return q;
```

```
}
```

Kết quả:

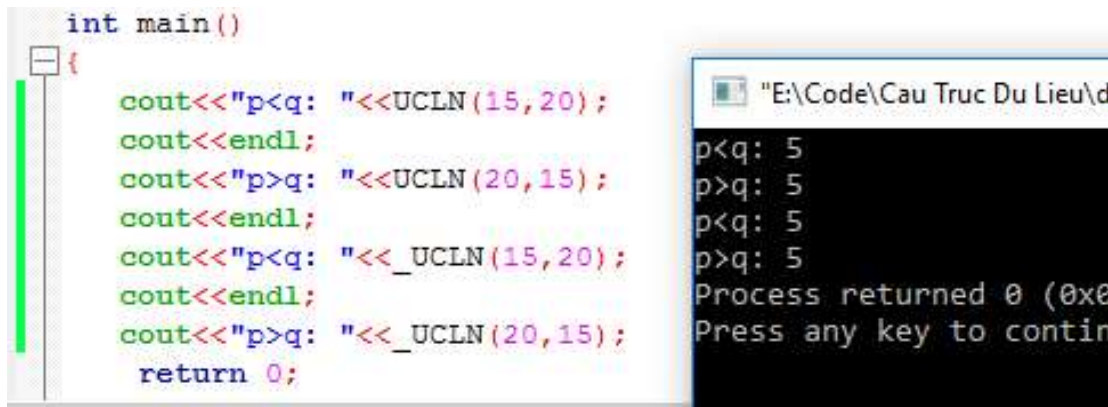
```
int main()
{
    cout<<UCLN(15,4008);
    cout<<endl;
    cout<<_UCLN(15,4008);
    return 0;
}
```



```
"E:\Code\Cau Truc
3
3
Process returned
Press any key to
```


Đặc điểm của giải thuật đệ quy ở bài này là: Gọi lại chính nó nhưng với không gian bài toán nhỏ hơn

Trường hợp $p < q$ thì **vẫn thế** (nên thay bằng **vẫn đúng**) không cần xử lý.

The image shows a C++ IDE with a code editor on the left and a console window on the right. The code in the editor is:

```
int main()
{
    cout<<"p<q: "<<UCLN(15,20);
    cout<<endl;
    cout<<"p>q: "<<UCLN(20,15);
    cout<<endl;
    cout<<"p<q: "<<_UCLN(15,20);
    cout<<endl;
    cout<<"p>q: "<<_UCLN(20,15);
    return 0;
}
```

The console window shows the output:

```
p<q: 5
p>q: 5
p<q: 5
p>q: 5
Process returned 0 (0x0)
Press any key to continue
```

Bài 4:

Nêu rõ các bước thực hiện việc gọi hàm `chuyen(3 ,A ,B , C)`; trong bài toán tháp Hà Nội.

Giả sử ta quy định A là cột nguồn, B là cột trung gian, C là cột đích

Chúng ta sẽ chia bài toán thành các bài toán nhỏ hơn. Cụ thể các bạn có thể xem video hướng dẫn tại đây:

https://youtu.be/xCAF_TZUCB8?list=PLZEIt444jqpC4og_M27gfYCE3t2u4PPDf

Việc gọi hàm `chuyen(3 ,A ,B , C)`;

```
#include <iostream>
```

```
using namespace std;
```

```
void chuyen(int n, char nguon, char trunggian, char dich)
```

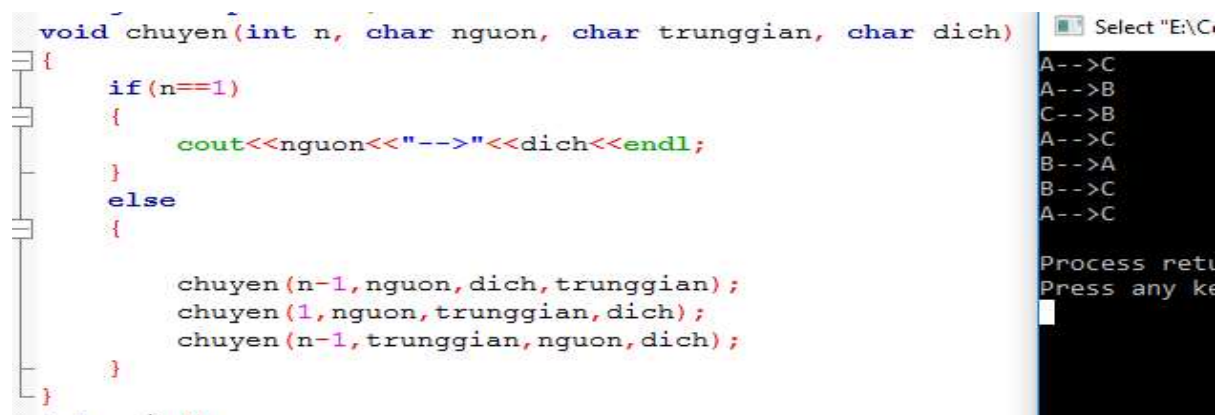
```
{
```

```
    if(n==1)
```

```

{
    cout<<nguồn<<"-->"<<dich<<endl;
}
else
{
    chuyen(n-1,nguồn,dich,trunggian);
    chuyen(1,nguồn,trunggian,dich);
    chuyen(n-1,trunggian,nguồn,dich);
}
}
int main()
{
    chuyen(3,'A','B','C');
    return 0;
}

```



The screenshot shows a C++ IDE with the following code in the editor:

```

void chuyen(int n, char nguồn, char trunggian, char dich)
{
    if (n==1)
    {
        cout<<nguồn<<"-->"<<dich<<endl;
    }
    else
    {
        chuyen(n-1,nguồn,dich,trunggian);
        chuyen(1,nguồn,trunggian,dich);
        chuyen(n-1,trunggian,nguồn,dich);
    }
}

```

To the right, a terminal window displays the output of the program:

```

A-->C
A-->B
C-->B
A-->C
B-->A
B-->C
A-->C
Process returned
Press any key

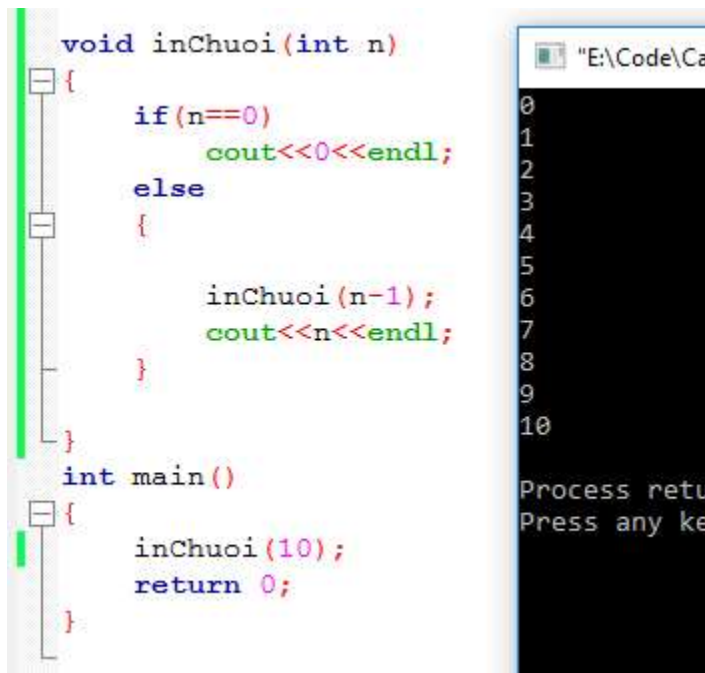
```

Bài 5:

Viết một hàm đệ quy và một hàm lặp thực hiện việc in ngược chuỗi “PASCAL” thì in ra là “LACSAP”

Để làm được bài này các bạn xem qua ví dụ sau:

Ở đây chúng ta chỉ **sét** $n \geq 0$

The image shows a C++ code editor on the left and a terminal window on the right. The code defines a recursive function 'inChuoi' that prints numbers from 0 to n in reverse order. The main function calls 'inChuoi(10)'. The terminal window shows the output: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, followed by the prompt 'Press any key'.

Chúng ta thấy rằng nếu $n = 0$ thì in ra 0 còn nếu n lớn hơn 0 thì gọi lại hàm **inChuoi** với tham số là $n-1$ rồi sau đó in ra n

Vậy chúng ta thử với $n = 2$, $n > 0$ nên nó sẽ gọi hàm **inChuoi(1)** hàm **inChuoi(1)** được gọi với $n=1$ tiếp tục lại lớn hơn 0 nó sẽ rơi vào trường hợp $n=0$ và chạy câu lệnh **cout<<0<<endl;** để in

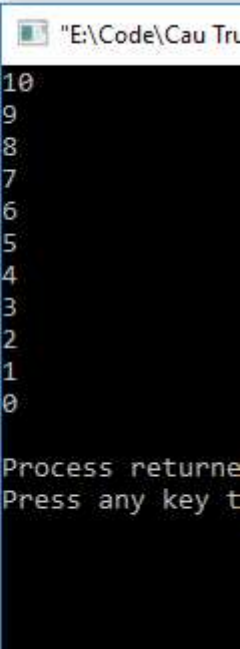
0 ra màn hình.

Sau đó trở lại hàm **inChuoi(1)** và in 1 ra màn hình, tiếp tục trở lại hàm **inChuoi(2)** và in 2 ra màn hình vậy ta được dãy số **0,1,2**

Các bạn sửa đoạn chương trình trên một chút

```
void inChuoi(int n)
{
    if(n==0)
        cout<<0<<endl;
    else
    {
        cout<<n<<endl;
        inChuoi(n-1);
    }
}

int main()
{
    inChuoi(10);
    return 0;
}
```



Cụ thể các bạn sửa như sau cho dòng **cout<<n<<endl;** lên trước lệnh gọi hàm **inChuoi(n-1);** chạy sẽ cho kết quả dãy số bị in ngược như hình trên.


Chúng ta lại phân tích một chút giả sử ở đây mình dùng **n= 2** cho nó bé bé để chúng ta tính bằng tay.

Ở **main** chúng ta gọi **inChuoi(2);** vậy hàm **inChuoi(2)** được truyền tham số là **n =2**, và tất nhiên **n!=0** nên sẽ chạy vào khối lệnh **else** gặp câu lệnh **cout<<n<<endl;** nó sẽ in ra màn hình **2** sau đó gọi lại hàm **inChuoi(n-1);** với tham số lúc này nhỏ hơn **inChuoi(2-1);** sau đó khi hàm này được gọi nó lại chạy hàm **inChuoi(1);** và in ra **1** cứ tương tự như vậy in ra được dãy 2,1,0

Lợi dụng đặc điểm này chúng ta sẽ in ra chuỗi đảo ngược như sau.

```
void inChuoi(int n, char a[])
{
    if(n==0)
        cout<<a[0];
    else
    {
        cout<<a[n];
        inChuoi(n-1, a);
    }
}

int main()
{
    char a[] = "PASCAL";
    inChuoi(strlen(a)-1, a);
    return 0;
}
```



Code:

```
#include <iostream>
#include <string.h>
using namespace std;
void inChuoi(int n, char a[])
{
    if(n==0)
        cout<<a[0];
    else
    {
        cout<<a[n];
        inChuoi(n-1, a);
    }
}
```

```

    }

}

int main()
{
    char a[] = "PASCAL";
    inChuoi(strlen(a)-1,a);
    return 0;
}

```

Bài 6: viết hàm đệ quy thực hiện việc đếm số chữ số của một số nguyên

```

#include <iostream>
#include <string.h>
using namespace std;
int demSo(int n)
{
    if(n<10&& n>-10)
    {
        return 1;
    }
    else
    {
        return demSo(n/10) + 1;
    }
}

```

```

    }
}

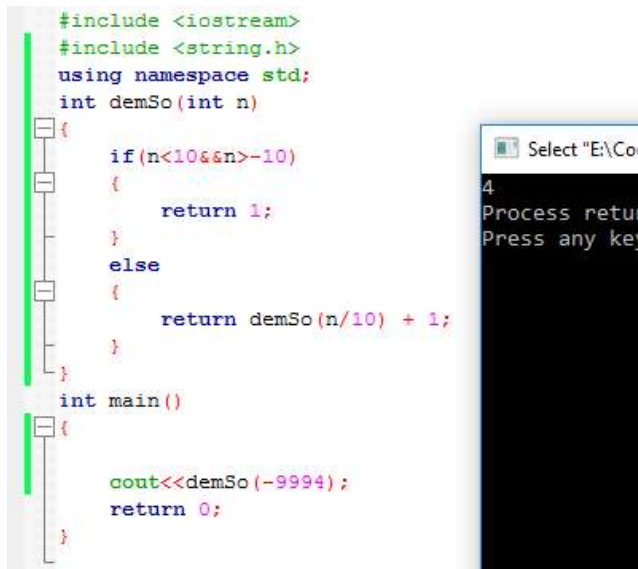
int main()
{

    cout<<demSo(-9994);

    return 0;

}

```

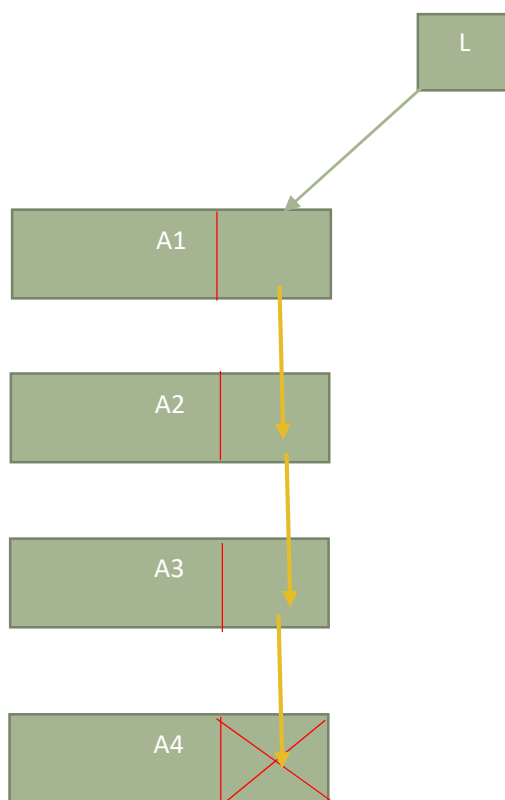


PHẦN 2: DANH SÁCH

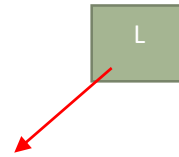
Ở phần này chúng ta sẽ ôn tập về các loại danh sách, bao gồm, danh sách liên kết đơn, danh sách liên kết kép, tôi không đề cập đến danh sách móc lồi vòng vì khi các bạn hiểu rõ các loại danh sách trên thì cài đặt danh sách móc lồi vòng rất đơn giản.

Bài 1: Cho danh sách liên kết đơn với nút đầu được trỏ bởi con trỏ L như hình vẽ

Vẽ hình mô tả trạng thái của danh sách trong quá trình hình thành danh sách như hình dưới, xuất phát từ danh sách rỗng



Khởi tạo danh sách rỗng, cho con trỏ L trỏ vào NULL

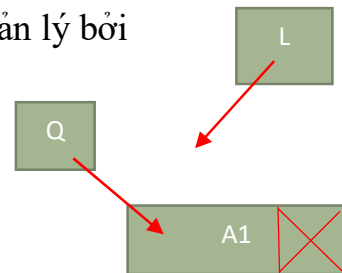


Ta thêm phần tử đầu tiên vào danh sách trước hết chúng ta phải tạo ra một phần tử sau đó ta đổ dữ liệu và trong nút đó, phần tử này được quản lý bởi

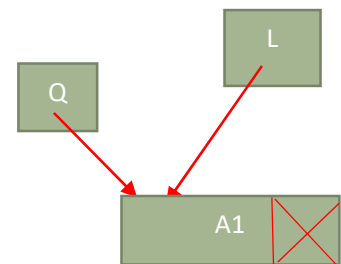
con trỏ q, và trường next của nút này ta gán bằng null

Vì là phần tử đầu tiên của danh sách vậy nên chúng ta

Chỉ cần gán $L = Q$

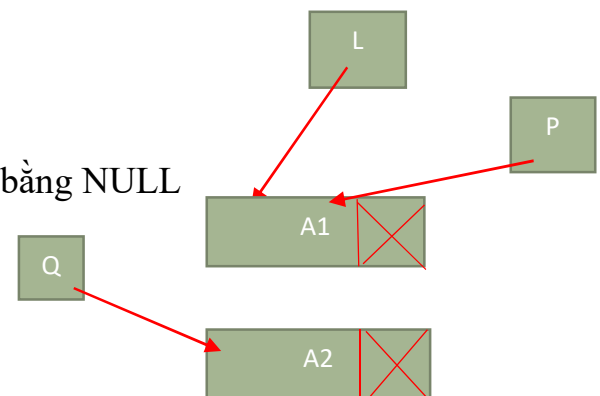


vậy chúng ta đã thêm được phần tử đầu tiên vào danh sách



bây giờ để thêm phần tử thứ 2 vào danh sách chúng ta cũng cần phải tạo ra một nút mới để chứa dữ liệu

chúng ta đổ dữ liệu vào nút này, gán cho trường next bằng NULL



tiếp theo chúng ta sẽ dùng một con trỏ p để tìm ra nút cuối cùng của danh sách

Chúng ta cứ lặp lại thao tác như vậy để thêm các phần tử vào danh sách.

- *Khai báo cấu trúc dữ liệu danh sách*

```
struct SinhVien
```

```
{  
    string msv;  
    string hoten;  
    int namsinh;  
    float diem;  
};
```

```
struct Node
```

```
{  
    SinhVien data;  
    Node *pNext;  
};
```

- *Tạo mới danh sách gồm 10 phần tử, nhập từ bàn phím.*

```
#include <iostream>
```

```
#include <stdio.h>
```

```
using namespace std;
```

```
int soSvGioi = 0;
```

```
int soSvKha = 0;
```

```
int soSvTB = 0;
```

```

int soSvYeu = 0;
struct SinhVien
{
    string msv;
    string hoten;
    int namsinh;
    float diem;
};

struct Node
{
    SinhVien data;
    Node *pNext;
};

typedef struct Node *TRO;
TRO l;

void create(TRO &l)
{
    l = NULL;
}

void nhapSinhVien(SinhVien sv)
{
    cout<<"Nhap MSV: "<<endl;

```

```

        fflush(stdin);
        getline(cin,sv.msv);
        cout<<"Nhap Ho Ten: "<<endl;
        fflush(stdin);
        getline(cin,sv.hoten);
        cout<<"Nhap Nam Sinh: "<<endl;
        cin>>sv.namsinh;
        cout<<"Nhap Dien: "<<endl;
        cin>>sv.diem;
    }
void xuatSinhVien(SinhVien sv)
{
    cout<<"MASV      : "<<sv.msv<<endl;
    cout<<"HO TEN    : "<<sv.hoten<<endl;
    cout<<"NAM SINH: "<<sv.namsinh<<endl;
    cout<<"DIEM      : "<<sv.diem<<endl;
    cout<<endl;
}
void add_last(TRO &l, SinhVien x)
{
    TRO q = new Node;
    q->data = x;
    q->pnext = NULL;

```

```

    if(l==NULL)
    {
        l = q;
    }
    else
    {
        TRO p = l;
        while(p->pnext!=NULL)
        {
            p = p->pnext;
        }
        p->pnext = q;
    }
}

void add_fist(TRO &l, SinhVien x)
{
    Node *q = new Node;
    q->data = x;

    if(l==NULL)
    {
        q->pnext = NULL;
        l = q;
    }
}

```

```

    }
    else
    {
        q->pnext = l;
        l = q;
    }
}

```

```

void themVaoViTriK(TRO &l, int k, SinhVien x)
{
    if(k==0)
    {
        add_fist(l,x);
    }
    Node *p = l;
    Node *q = new Node;
    q->data = x;
    int i = 0;
    while(p->pnext!=NULL)
    {
        i++;
        if(k==i)
        {

```

```

        q->pnext = p->pnext;
        p->pnext = q;
        return;

    }

    p=p->pnext;
}

add_last(l,x);

}

void duyetDanhSach(TRO l)
{
    while(l!=NULL)
    {
        if(l->data.diem>8.0)
        {
            soSvGioi++;
        }
        else if(l->data.diem>7)
        {
            soSvKha++;
        }
    }
}

```



```

        else if(l->data.diem>5)
        {
            soSvTB++;

        }
        else
        {
            soSvYeu++;
        }
        xuatSinhVien(l->data);
        l = l->pnext;
    }
}

```

```

void xoaFist(TRO &l)
{
    TRO q = l;
    l = l->pnext;
    delete q;
}

```

```

void xoaPhanTuThuK(TRO &l, int k)
{

```

```

int i= 0;
Node * p = l;
while(p->pnext!=NULL)
{
    i++;
    if(i==k)
    {
        Node *q = p->pnext;
        p->pnext = p->pnext->pnext;
        delete q;
    }
    p=p->pnext;
}
}

int main()
{
    TRO l;
    SinhVien a;
    a.diem = 10;
    a.hoten = "son";
    a.msv = "1010";
    a.namsinh = 1997;
    create(l);
}

```

```

for(int i =0;i<10;i++)
{
    a.namsinh = i;
    add_last(l,a);
    //add_fist(l,a);
}
duyetDanhSach(l);
//xoaFist(l);
//xoaPhanTuThuK(l,2);
a.hoten = "THEM VAO";
themVaoViTriK(l,1,a);
cout<<"\\n XOA"<<endl;
duyetDanhSach(l);
return 0;
}

```

<p>Đệ quy:</p> <pre> #include<conio.h> #include<stdio.h> #include<iostream> using namespace std; void nhap(int &n) { cout<<"Moi ban nhap n:"; cin>>n; } </pre>	<p>Chuyen thap HN</p> <pre> int chuyen(int n, int a, int b, int c) { if(n==1) cout<<"\\n chuyen dia tu cot"<<a<<"sang cot "<<b; else { chuyen(n-1,a,c,b); chuyen(1,a,b,c); } } </pre>
---	--

```

void nhap(int &m,int &n)
{
    cout<<" Moi ban nhap
m:"; cin>>m;
    cout<<"Moi ban nhap
n:"; cin>>n;
}
int Acker(int m,int n)
{
    if(m==0) return n+1;
    if(n==0) return
Acker(m-1,1);
    else return
Acker(m-1,Acker(m,n-1));
}
int GiaiThua(int n)
{
    if(n==0) return 1;
    else return n*GT(n-1);
}
int FI(int n)
{
    if(n<=2) return 1;
    return FI(n-2)+FI(n-1);
}
int demsochuso(int n)
{
    if(n<10) return 1;
    return
demsochuso(n/10)+1;
}

```

```

chuyen(n-1,c,b,a);
    }

int main()
{
    int m,n;
    //nhap(n);
    nhap(m,n);
    //cout<<" Giai thua
cua"<<n<<" la:"<<GT(n);
    //cout<<"\n Ket qua Fibo
cua "<<n<<"la:"<<FI(n);
    //cout<<"\n Dem so chu so
cua "<<n<<"
la:"<<demsochuso(n);
    //chuyen(n,1,2,3);
    cout<<"Ket qua cua Acker
la:"<<A(m,n);
    return 0;
}

```

<p>Sắp xếp và tìm kiếm:</p> <pre> #include<conio.h> #include<stdio.h> #include<iostream> #include<iomanip> #include<string.h> #include<math.h> using namespace std; struct node { int key; node *left,*right; }*t; void insert(int x, node *&t) { if(t==NULL) { t=new node; t->key=x; t->left=NULL; t->right=NULL; } else if(x<t->key) insert(x,t->left); else insert(x,t- >right); } void nhap cay(node *&t) { int n,x; t=NULL; cout<<"Nhap so nut: "; cin>>n; for(int i=1;i<=n;i++) { </pre>	<pre> void nhap mang(int a[],int &n) { do { cout<<"Nhap so phan tu cua mang:"; cin>>n; } while(n<=0 n>=100); for(int i=0;i<n;i++) { cout<<" Nhap phan tu thu"<<i+1<<": "<<"a["<<i<<"]="; cin>>a[i]; } } void hien mang(int a[],int n) { for(int i=0;i<n;i++) cout<<a[i]<<" "; } void sx_chontang(int a[],int n) { int i,j,tg,m; for(i=0;i<n;i++) { m=i; for(j=i+1;j<n;j++) if(a[j]<a[m]) m=j; if(m!=i) { tg=a[i]; a[i]=a[m]; a[m]=tg; </pre>
---	--

```

        cout<<"Nhap nut thu
"<<i<<" ";
        cin>>x;
insert(x,t);
    }
}
void hien cay( node *&t)
{
    if(t!=NULL)
    {
        hien(t->left);
        cout<<t->key<<" ";
        hien(t->right);
    }
}

void chentang(int
a[],int n)
{
    int i,j,tg;
    for(i=1;i<n;i++)
    {
        j=i-1;
        tg=a[i];
while((j>=0)&& a[j]>tg)
        {
            a[j+1]=a[j];
            j--;
        }
        a[j+1]=tg;
    }
}
void noibottang(int a[],int
n)
{
    int i,j,tg;
    for(i=0;i<n-1;i++)
        for(j=n-1;j>i;j--)

```

```

    }
}

void phandoan(int l,int
r,int a[])
{
    if(l<r)
    {
        int i,j,x,tg;
        x=a[(l+r)/2];
        i=l;j=r;
        while(i<=j)
        {
            while(a[i]<x) i++;
            while(a[j]>x) j--;
            if(i<=j)
            {
                tg=a[i];
                a[i]=a[j];
                a[j]=tg;
                i++;j--;
            }
        }
        phandoan(l,j,a);
        phandoan(i,r,a);
    }
}

```

```

        if(a[j]<a[j-1])
        {
            tg=a[j];
            a[j]=a[j-1];
            a[j-1]=tg;
        }
    }
}
void noibotcaitientang(int
a[],int n)
{
    int i=0,j,tg,d;
    while(i<n-1)
    {
        d=n;
        for(j=n-1;j>i;j--)
            if(a[j]<a[j-1])
            {
                tg=a[j];
                a[j]=a[j-1];
                a[j-1]=tg;
                d=j;
            }
        i=d;
    }
}

```

Vun đồng

```

void push(int i,int a[],int
n)
{
    int j,tg=a[i];
    while(i<=n/2-1)
    {
        j=2*i+1;
        if((j<n-
1)&&(a[j]<a[j+1]))
            j++;
        if(a[j]>tg)
        {

```

```

void mpass(int a[],int b[],int
n,int l)
{
    int i,j;    i=-1;
    while(i+2*l<n)
    {
        merge(a,i+1,i+1,i+2*l,b);
        i+=2*l;
    }
    if(i+1<n-1)
        merge(a,i+1,i+1,n-1,b);
    else for(j=i+1;j<n;j++)
        b[j]=a[j];
}
void mergesort(int a[], int n)
{
    int l,b[100];
    l=1;

```

<pre> a[i]=a[j]; i=j; } else n=i; } a[i]=tg; } void vundong(int a[],int n) { int i,tg; for(i=n/2-1;i>=0;i--) push(i,a,n); for(i=n-1;i>0;i--) { tg=a[0]; a[0]=a[i]; a[i]=tg; push(0,a,i); } } Hòa Nhập void merge (int a[],int d1,int c1,int c2,int b[]) { int i,j,k; i=d1;k=d1;j=c1+1; while((i<=c1)&&(j<=c2)) if(a[i]<a[j]) b[k++]=a[i++]; else b[k++]=a[j++]; while(i<=c1) b[k++]=a[i++]; while(j<=c2) b[k++]=a[j++]; } </pre>	<pre> while(l<n) { mpass(a,b,n,l); l=2*l; mpass(b,a,n,l); l=2*l; } } void mpass2(int a[],int b[],int n,int &d) { int c1,c2,i=-1,j; d=0; do{ c1=i+1; while((a[c1]<=a[c1+1])&&(c1<n- 1)) c1++; if(c1<n-1) { c2=c1+1; while((a[c2]<=a[c1+1])&&(c2<n- 1))c2++; merge(a,i+1,c1,c2,b); i=c2;d=1; } else { for(j=i+1;j<n;j++) b[j]=a[j];i=n; } while(i<n); } int tk_nhiphan(int x,int a[],int n) </pre>
---	--


```

void mergesort2(int a[],
int n)
{
    int b[100],d;
    do
    {
        mpass2(a,b,n,d);
        mpass2(b,a,n,d);
    }
    while(d!=0);
}

```

Tìm kiếm

```

int TuanTu_cs(int x,int
a[],int n)
{
    int i=0;
    while(i<n&& a[i]!=x) i++;
    if(i<n) return 1;
    return 0;
}
Int TT_CS_CT(int x,int
a[],int n)
{
    int i=0;
    a[n]=x;
    while(a[i]!=x) i++;
    if(i<n) return 1;
    return 0;
}
int tktttdasap(int x,int
a[],int n)
{
    int i=0;
    a[n]=x;
    while(a[i]<x) i++;
    if(i<n&&a[i]==x) return
1;

```

```

{
    int l=0,r=n-1,m;
    while(l<=r)
    {
        m=(l+r)/2;
        if(a[m]==x) return 1;
        if(a[m]>x) r=m-1;
        else l=m+1;
    }
    return 0;
}

int tknp_dq(int x,int a[],int
l,int r)
{
    if(r<l) return 0;
    int m=(l+r)/2;
    if(a[m]==x) return 1;
    if(a[m]>x) return
tknp_dq(x,a,l,m-1);
    return tknp_dq(x,a,m+1,r);
}
int tk_caynp(int x,node *t)
{
    while(t!=NULL)
    {
        if(t->key==x) return
1;
        if(t->key>x) t=t-
>left;
        else t=t->right;
    }
    return 0;
}
int tk_cay_npdq(int x, node
*t)
{
    if(t==NULL) return 0;
    if(t->key==x) return 1;

```

<pre> return 0; } int tuantu_dequy(int x,int a[],int n) { if(n==0) return 0; if(a[n-1]==x) return 1; else return taunt_dequy(x,a,n-1); } int main() { int a[100],b[100],n,x; //nhap(a,n); nhap(t); cout<<"Mang vua nhap la: \n"; //hien(a,n); hien(t); //chontang(a,n); //chentang(a,n); //noibottang(a,n); //noibotcaitientang(a,n); //phandoan(0,n-1,a); //vundong(a,n); //mergesort(a,n); //mergesort2(a,n); //cout<<"\n Mang sau khi sap xep la:\n"; // hien(t); cout<<"Nhap x:"; cin>>x; </pre>	<pre> if(t->key>x) return tk_cay_npdq(x,t->left); return tk_cay_npdq(x,t- >right); } NHAP n Phần Tử móc nối đơn void nhap_n(dssv*&l, int n) { dssv *q; l=new dssv; q=l; cout<<"Nhap ma sinh vien:";fflush(stdin);gets(l- >ptu.masv); cout<<"Nhap ho ten:";fflush(stdin); gets(l->ptu.hten); cout<<"Nhap diem trung binh:"; cin>>l->ptu.dtb; int d=1; while(d<n) { q->next=new dssv; q=q->next; cout<<"Nhap ma sinh vien:"; fflush(stdin);gets(q- >ptu.masv); cout<<"Nhap ho ten:"; </pre>
--	--

<pre> //int v=timkiemlap(x,a,n); //int v=timkiemcaitien(x,a,n); //int v=tktttdasap(x,a,n); //int v=timkiemdequy(x,a,n); //int v=tknp(x,a,n); int v=tkcnp(x,t); //int v=tkcnpdq(x,t); if(v==1) cout<<"Tim thay x"; else cout<<"Khong tim thay x"; return 0; } </pre>	<pre> fflush(stdin);gets(q- >ptu.hten); cout<<"Nhap diem trung binh:"; cin>>q->ptu.dtb; d++; } q->next=NULL; } </pre>
---	--

Móc nối đơn:

```
#include<iostream>
#include<conio.h>
#include<stdio.h>
#include<string.h>
#include<iomanip>
#include<stdlib.h>
using namespace std;
struct svien
{
    char masv[10];
    char hten[20];
    float dtb;
};
struct dssv
{
    svien ptu;
    dssv *next;
}*l;
void nhap(dssv *&l)
{
    dssv *q;
    char s[10];
    l=new dssv;    q=l;
    cout<<"Nhap ma sinh
vien:";fflush(stdin);gets(l-
>ptu.masv);
    cout<<"Nhap ho
ten:";fflush(stdin);gets(l-
>ptu.hten);
    cout<<"Nhap diem trung
binh:";cin>>l->ptu.dtb;
    cout<<"Nhap ma sinh
vien:";fflush(stdin);gets(s);
    while(strcmp(s,""))
    {
        q->next=new dssv;
        q=q->next;
```

```
void hien(dssv *l)
{
    cout<<setw(40)<<"DANH
SACH SINH VIEN \n";
    cout<<setw(10)<<"Ma
sv"<<setw(20)<<"ho
ten"<<setw(20)<<"diem
tb"<<"\n";
    while(l!=NULL)
    {
        cout<<setw(10)<<l-
>ptu.masv<<setw(20)<<l-
>ptu.hten<<setw(20)<<l-
>ptu.dtb<<"\n";
        l=l->next;
    }
}
void sapxepgiam(dssv *&l)
{
    dssv *p,*q;
    svien tg;
    if(l!=NULL)
    {
        p=l;
        while(p->next!=NULL)
        {
            q=p->next;
            while(q!=NULL)
            {
                if(p-
>ptu.dtb<q->ptu.dtb)
                {
                    tg=q-
>ptu;
                    q-
>ptu=p->ptu;
                    p-
>ptu=tg;
```

```

        strcpy(q-
>ptu.masv,s);
        cout<<"Nhap ho
ten:";fflush(stdin);gets(q-
>ptu.hten);
        cout<<"Nhap diem
trung binh:";cin>>q->ptu.dtb;
        cout<<"Nhap ma sinh
vien:";fflush(stdin);gets(s);
    }
    q->next=NULL;
}
void xoadau(dssv *&l)
{
    if(l!=NULL)
    {
        dssv *p=l;
        l=l->next;
        delete p;
    }
}
void xoacuo(i(dssv *&l)
{
    if(l!=NULL)
    if(l->next==NULL)
    {
        delete l;
        l=NULL;
    }
    else
    {
        dssv *p=l,*q;
        while(p->next-
>next!=NULL)
            p=p->next;
        q=p->next;
        p->next=NULL;
        delete q;
    }
}

```

```

    }
    q=q->next;
}
p=p->next;
}
}
}
}
void xoasaptang(dssv *&l)
{
    dssv *q;
    while(l!=NULL&&l-
>ptu.dtb<5.0)
    {
        q=l;l=l->next;
        delete q;
    }
}
void loailapma(dssv *&l)
{
    if(l!=NULL)
    {
        dssv *p=l,*q;
        while(p->next!=NULL)
        {
            q=p;
            while(q-
>next!=NULL)
            if(!strcmp(p->ptu.masv,q-
>next->ptu.masv))
            {
                dssv *r=q-
>next;

```

<pre> } void xoasapgiam(dssv *&l) { dssv *p; if(l!=NULL) { if(l->ptu.dtb<5.0) l=NULL; else { p=l; while(p- >next!=NULL&& p->next- >ptu.dtb>=5.0) p=p->next; p->next=NULL; } } } </pre>	<pre> q->next=r- >next; delete r; } else q=q->next; if(p- >next!=NULL) p=p->next; } } void xoa_k(int k, dssv *&l) { dssv*q=l, *p; int d=0; while(d<k-2) {q=q->next;d++;} p=q->next; q->next=p->next; delete p; } </pre>
<pre> void chendau(dssv *&l) { dssv *p; p=new dssv; cout<<"Nhap ma sinh vien:";fflush(stdin);gets(p- >ptu.masv); cout<<"Nhap ho ten:";fflush(stdin);gets(p- >ptu.hten); cout<<"Nhap diem trung binh:";cin>>p->ptu.dtb; p->next=l; l=p; } void chencuoi(dssv *&l) { </pre>	<pre> void chentang(dssv *&l) { dssv *p=new dssv; cout<<"Nhap ma sinh vien:";fflush(stdin);gets(p- >ptu.masv); cout<<"Nhap ho ten:";fflush(stdin);gets(p- >ptu.hten); cout<<"Nhap diem trung binh:";cin>>p->ptu.dtb; if(l==NULL p- >ptu.dtb<=l->ptu.dtb) { p->next=l; l=p; } } </pre>

```

    dssv *p;
    p=new dssv;
    cout<<"Nhap ma sinh
vien:";fflush(stdin);gets(p-
>ptu.masv);
    cout<<"Nhap ho
ten:";fflush(stdin);gets(p-
>ptu.hten);
    cout<<"Nhap diem trung
binh:";cin>>p->ptu.dtb;
    p->next=NULL;
    if(l==NULL)
    l=p;
    else
    {
        dssv *q=l;
        while(q->next!=NULL)
            q=q->next;
        q->next=p;
    }
}

```

```

    else
    {
        dssv *q=l;
        while(q-
>next!=NULL&&q->next-
>ptu.dtb<p->ptu.dtb)
            q=q->next;
        p->next=q->next;
        q->next=p;
    }
}
void chen_k(int k, dssv *&l)
{
    dssv *p=new dssv, *q=l;
    cout<<"Nhap ma sinh
vien:";fflush(stdin);gets(p-
>ptu.masv);
    cout<<"Nhap ho
ten:";fflush(stdin);gets(p-
>ptu.hten);
    cout<<"Nhap diem trung
binh:";cin>>p->ptu.dtb;
    int d=0;
    while(d<k-1)
        {q=q->next;d++;}
    p->next=q->next;
    q->next=p;
}

```

```

void chengiam(dssv *&l)
{
    dssv *p=new dssv;
    cout<<"Nhap ma sinh
vien:";fflush(stdin);gets(p-
>ptu.masv);
    cout<<"Nhap ho
ten:";fflush(stdin);gets(p-
>ptu.hten);
    cout<<"Nhap diem trung
binh:";cin>>p->ptu.dtb;
    if(l==NULL||p-
>ptu.dtb>=l->ptu.dtb)
    {
        p->next=l;
        l=p;
    }
    else
    {
        dssv *q=l;
        while(q-
>next!=NULL&&q->next-
>ptu.dtb>p->ptu.dtb)
            q=q->next;
        p->next=q->next;
        q->next=p;
    }
}
void daochieu(dssv *&l)
{
    if(l!=NULL)
    {
        dssv *p=l,*q=NULL;
        while(l->next!=NULL)
        {
            l=l->next;
            p->next=q;
            q=p;
        }
    }
}

```

```

void nhap_makhongtrung(dssv
*&l)
{
    dssv *q;
    char s[10];
    l=new dssv;
    q=l;
    cout<<"Nhap ma sinh
vien:";fflush(stdin);gets(l-
>ptu.masv);
    cout<<"Nhap ho
ten:";fflush(stdin);gets(l-
>ptu.hten);
    cout<<"Nhap diem trung
binh:";cin>>l->ptu.dtb;
    cout<<"Nhap ma sinh
vien:";fflush(stdin);gets(s);
    while(strcmp(s,""))
    {
        dssv *p=l;
        while(p!=q&&strcmp(p-
>ptu.masv,s)) p=p->next;
        if(strcmp(p-
>ptu.masv,s))
        {
            q->next=new dssv;
            q=q->next;
            strcpy(q-
>ptu.masv,s);
            cout<<"Nhap ho
ten:";fflush(stdin);gets(q-
>ptu.hten);
            cout<<"Nhap diem
trung binh:";cin>>q->ptu.dtb;
        }
        else cout<<"Ma trung
\n";
        cout<<"Nhap ma
sv:";fflush(stdin);gets(s);
    }
}

```


<pre> p=l; } l->next=q; } }</pre>	<pre> } q->next=NULL; }</pre>
---	---