

```
////////////////////////////////////////////////////////////////  
// Chương trình dùng DSLK đơn (kiểu hướng đối tượng) để quản lý nhiều phân tử  
// có cấu trúc khác nhau thông qua dữ liệu kiểu con trỏ void để quản lý  
// ngày 22 tháng 04 năm 2004 - @ by Phạm Thế Bảo  
// Khoa Toán – Tin học trường ĐHKHTN Tp. Hồ Chí Minh  
////////////////////////////////////////////////////////////////
```

```
#include<iostream.h>  
#include<stdio.h>  
#include<conio.h>  
#include<string.h>
```

```
struct NV{  
    int luong;  
    char *name;  
};
```

```
class Student{  
    private:  
        int code;  
        char *name;  
    public:  
        Student(int c,char* st){code=c;name = st;}  
        ~Student(){code=-1;}  
        int getCode(){return code;}  
        char *getName(){return name;}  
};
```

```
class TagNode{  
friend class LinkedList;  
    private:  
        void *data;  
        char kind;  
        TagNode *Next;  
    public:  
        TagNode();  
        TagNode(void *,char);  
        ~TagNode();  
};
```

```
typedef TagNode *Node;  
class LinkedList{
```

```
private:
    Node Head,Tail;
public:
    LinkedList();
    ~LinkedList();
    int insertNodeAtHead(void *d,char t);
    void print();
};
/////////////////////////////////////////////////////////////////
// code của các phương thức trong 02 lớp TagNode và LinkedList
/////////////////////////////////////////////////////////////////
TagNode::TagNode(){
    kind=-1;
    data = Next = NULL;
}
TagNode::TagNode(void *a,char t){
    kind = t;
    data = a;
    Next = NULL;
}
TagNode::~~TagNode(){
    kind =-1;
    if(Next) delete Next;
}
LinkedList::LinkedList(){
    Head = Tail = NULL;
}
LinkedList::~~LinkedList(){
    if(Head)delete Head;
}
int LinkedList::insertNodeAtHead(void *d,char t){
    Node p = new TagNode(d,t);
    if(!p)return -1;
    if(!Head)Head = Tail = p;
    else{
        p->Next = Head;
        Head = p;
    }
    return 1;
}
void LinkedList::print(){
```



```
for(int i=12;i<20;i++){
    c = (void *) i;
    ll.insertNodeAtHead(c,0);
}

char *temp;
printf("Nhap 01 chuoi ky tu:");
gets(temp);
c = (void *) temp;
ll.insertNodeAtHead(c,1);

Student *temp1= new Student(1243,"I like coco");
c = (void *)temp1;
ll.insertNodeAtHead(c,2);

NV *a= new NV;;
a->luong = 300;
printf("Nhap ho ten:");
gets(a->name);
c = (void *) a;
ll.insertNodeAtHead(c,3);

for(i=65;i<100;i++){
    c = (void *) i;
    ll.insertNodeAtHead(c,4);
}

ll.print();
getch();
return 1;
}
```