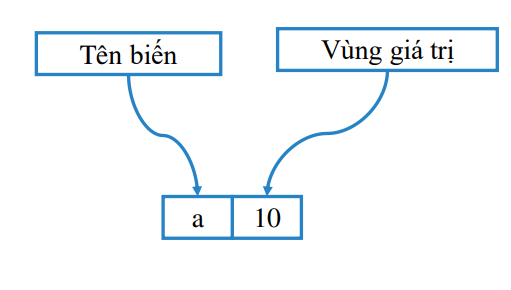
1. Dslk

* Biến tĩnh(static):

+ Khai báo:

<kiểu DL> tên biến;



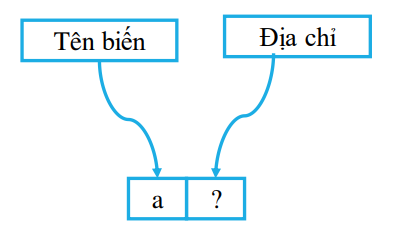
vd: int a; float y; char s[20];

a= 10; // gán giá trị cho a.

* Biến động:

+ khai báo: <kiểu DL> \*tên biến;

vd : int \*a; float \*y;



+ Thao tác:

* Cấp phát bộ nhớ: new.
* Giải phóng bộ nhớ: delete.

+ VD:

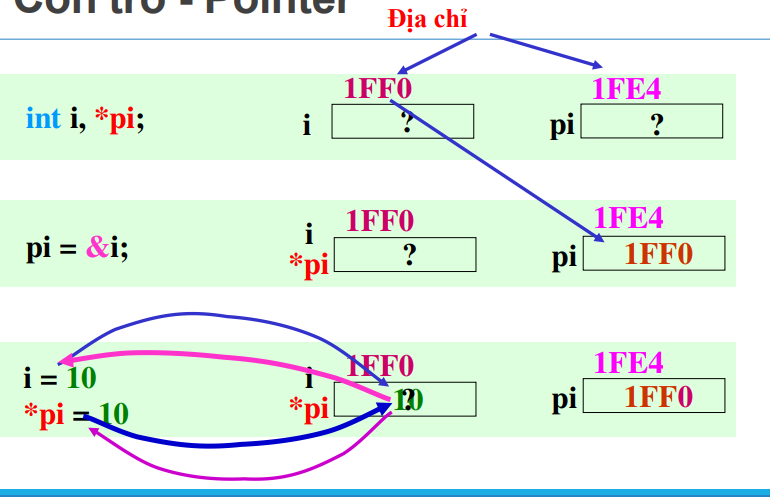
int \*a; // khai báo.

a= new int ; // cấp phát.

\*a= 10; // các thao tác (…)

delete a; // giải phóng

* Con trỏ(pointer):



* Các thao tác:

// hàm tạo Node

Node \*createNode(int x){

    Node \*p= new Node;

    if(p == NULL){

        printf("\n Khong du bo nho de cap nhat.");

        return NULL;

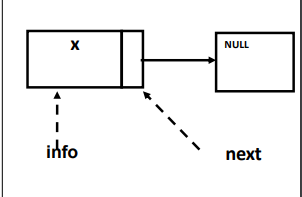
    }

    p->info= x;

    p->next = NULL;

    return p;

}



// hàm xóa Node

void deleteNode(Node \*&p){

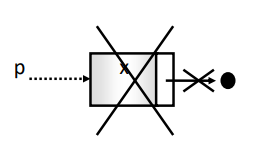
    if(p == NULL)

        return;

    p->next = NULL;

    delete p;

}



// khởi tạo dslk đơn

void createSlist(SList &l, int n){

    int x;

    Node \*q;

    l.head= l.tail = NULL;

    for(int i=1; i<= n; i++)

    {

        printf("\n Nhap pt thu %d la : ",i);

        scanf("%d",&x);

        q= createNode(x);

        if(q == NULL){

            printf("\n Khong du bo nho de cap phat.");

            getchar();

            return;

        }

        // nối p vào ds 1

        if(l.head == NULL)

            l.head= l.tail = q;

        else{

            l.tail->next = q;

            l.tail = q;

        }

    }

}

// thêm node vào đầu dslk

void addHeadSlist(SList &l, Node \*p){

    if(p == NULL)

        return ; // p  rỗng thì ko thêm

    else

        if(l.head == NULL) // ds rỗng

        l.head = l.tail= p;

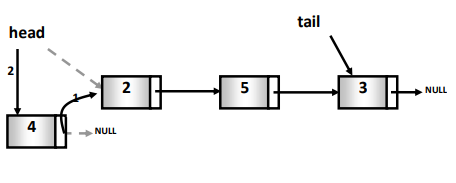
        else{

            p->next = l.head;

            l.head = p;

        }

}



// thêm node vào cuối dslk

void addTailSlist(SList &l, Node \*p){

    if( p == NULL)

        return;

    else

        if(l.head == NULL)

            l.head= l.tail = p;

        else{

            l.tail->next = p;

            l.tail = p;

        }

}

