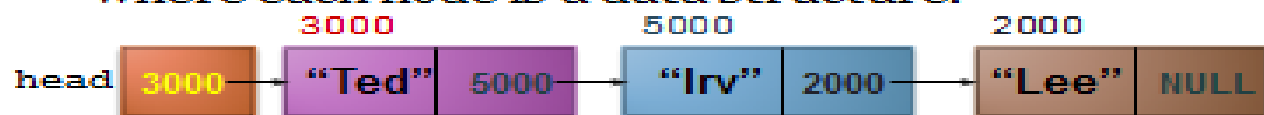


LINKED LIST

LINKED LIST

Introduction

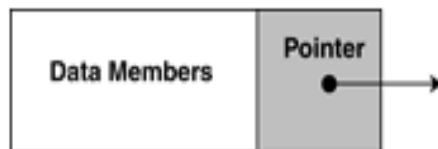
- A linked list is a series of connected *nodes*, where each node is a data structure.



- A linked list can grow or shrink in size as the program runs

The composition of a Linked List

- Each node in a linked list contains one or more members that represent data.
- In addition to the data, each node contains a pointer, which can point to another node.



Declaration of Linked List:

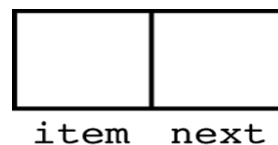
A node in a linked list is usually a struct •

```
struct Node
{
    int item;
    Node *next;
};
```

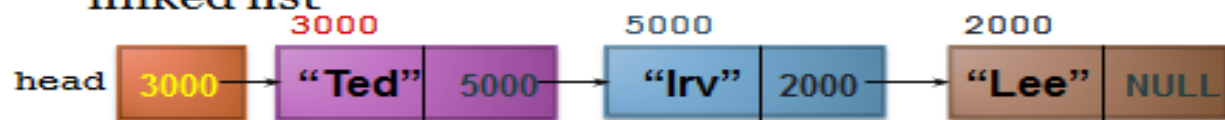
A node is dynamically allocated •

```
Node *p;
```

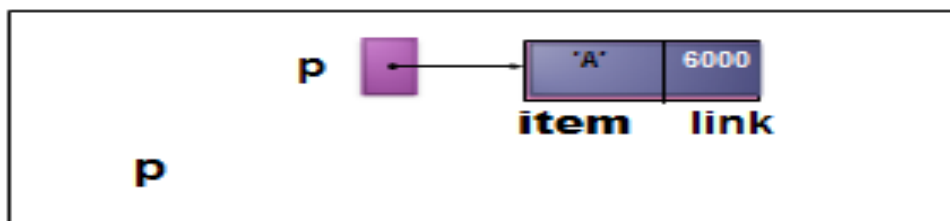
```
p = new Node;
```



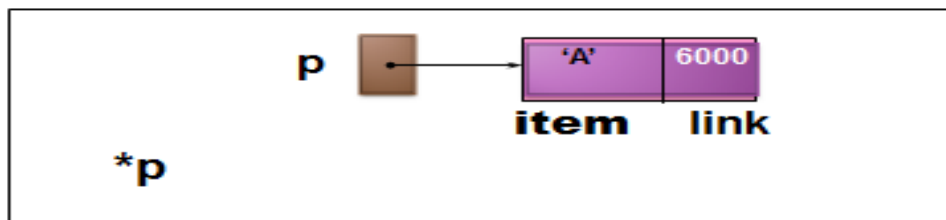
- The head pointer points to the first node in a linked list



- If head is *NULL*, the linked list is empty



p is a pointer to a node



$*p$ is the entire node pointed to by p

Ex1: write program to calculate the number of even and odd numbers in linked list.

```
#include<iostream.h>
struct nod{
int info ;
nod *next ;   //odd ,even
};

main()
{ int o=0,e=0;
nod *p,*q,*f;
f=new nod;   // أنشئ أول نود وتعتبر هذه هي رأس المتسلسلة ويجب أن يبقى على أول نود
cout<<"numbers: ";
cin>>f->info;
f->next=0;
p=f;
for(int i=1;i<=3;i++) // كل ما أنشئ نود جديدة أربطة بالنود السابقة وأقدم المؤشر
{
    q=new nod;
    cin>>q->info;
    q->next=0;
    p->next=q;
    p=q;
}
p=f;
while(p!=0)
{
    if(p-> info %2==0)
        o++;
    else e++;
    p=p->next;
} cout<<"odd= "<<e<<" even="<<o;
}
```

Ex2: write program to create linked list of char with 10 node then write function to add new node, function to delete node, and function to print the list.

```
#include<iostream.h>
```

```
struct node
```

```
{ int x;
```

```
node *nex;};
```

```
void add(node *f)
```

```
{ node *p,*q; p=f;
```

```
int c; cout<<"choose element\n"; cin>>c ;
```

```
while (p->x!=c) p=p->nex;
```

```
q=new node; cout<<"read node value\n";
```

```
cin>>q->x; q->nex=p->nex; p->nex=q;
```

```
}
```

```
void print(node *f)
```

```
{ node *p; p=f;
```

```
while( p!=0)
```

```
{ cout<<p->x; p=p->nex;}
```

```
}
```

```
void delet(node *f)
```

```
{ node *p,*q; p=f;
```

```
int c; cout<<"chose element\n"; cin>>c;
```

```
while( p->x!=c)
```

```
{q=p; p=p->nex; }
```

```
q->nex=p->nex;
```

```
}
```

```
void main()
```

```
{   node *f,*p,*q;
```

```
    cout<<"\n enter your list\n";
```

```
    f=new node;
```

```
    cin>>f->x;   f->nex=0;   p=f;
```

```
    for(int i=1;i<5;i++)
```

```
    { q=new node;
```

```
      cin>>q->x;   q->nex=0;
```

```
      p->nex=q;   p=q;}
```

```
    p=f;
```

```
    int y=0, c;
```

```
do
```

```
    {   cout<<"\n1= add\n";
```

```
        cout<<"2=print\n";
```

```
        cout<<"3=delete\n";
```

```
        int ch;      cin>>ch;
```

```
        switch(ch)
```

```
        {case 1:add(f); break;
```

```
          case 2: print(f);break;
```

```
          case 3:delet(f);
```

```
        }
```

```
    } while(y!=3);
```

```
}
```

