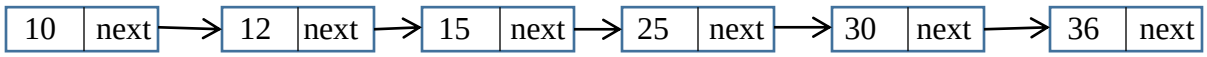
Linked list Examples

EX1: Consider the following linked list:



What will be the value of the following status:

Head->next->next->next->data

Answer: the value is 25.

EX2: Write program to create linked list consist of N node :

```
#include<iostream>
Using namespace std;

struct node{
  int data;
  node* next;
  }*head,*p,*q;

  void addnode(int m)
  {
    int value;
    while(m>0)
    {
      cout<<"enter value:\n";
      cin>>value;
      q=new node;
      q->data=value;
      q->next=NULL;
      if(head==NULL)
       { head=q; p=q; }
      else
       {
        p->next=q;
        p=p->next;
       }
       m=m-1;
     }
   }
/*--------------------------------*/
```

```
void display( )
 {
    p=head;
    cout<<"THE LIST:\n";
    while(p)
    {
     cout<<p->data<<endl;
     p=p->next;
    }
 }
/*----------------------------------------*/

 int main()
 {
   int n;

   cout<<"ENTER N:\n";
   cin>>n;
   addnode(n);
   display();
   return 0;
  }
```

EX3: write function to calculate the length of linked list:

```
int  length()
{
 int l=0;
 p=head;
 while(p)
  {
    l++;
    p=p->next;
  }
  return l;
}
/*----------------------------------*/
```

EX4: Write program to insert node in specific position :

```
#include<iostream>
Using namespace std;

struct node{
  int data;
  node* next;
  }*head,*p,*q;

/*----------------------------------------*/
 void creatnode(int value)
 {
      q=new node;
      q->data=value;
      q->next=NULL;

  }
 /*----------------------------------------*/

void insertfirst(int d)
{
q=new node;
q->data=d;
q->next=head;
head=q;
}
/*------------------------------*/

void addinpos(int po)
{
   int v;

     cout<<"ENTER VALUE:\n";
     cin>>v;
     if(head==NULL)
      { creatnode(v); head=q;}
     else
      if(po==0)
       insertfirst(v);
      else
      { p=head;
        for(int i=1;i<po-1;i++)
          p=p->next;

        creatnode(v);
        q->next=p->next;
        p->next=q;
      }}
```

```
/*----------------------------------------*/

 int main()
 {
   int n,pos;


   cout<<"ENTER N:\n";
   cin>>n;
   addnode(n);
   display();
   cout<<"ENTER POSITION :"<<endl;
   cin>>pos;
   addinpos(pos);
   display();
   return 0;
  }
```

Comparison between **array** and **linked list**:

	array	**Linked list**
1	Fixed length	Changeable length
2	Access elements by index	Access elements by traversal
3	Insertion /removable is costly	Insertion / removable efficient

استخدامات القوائم المرتبطة:

القوائم المتصلة على سبيل المثال تستخدم بشكل أساسي داخل البنوك ، فحساب كل شخص هو عبارة عن عقدة node فهي تستخدم لحفظ البيانات بشكل مؤقت وتتعامل مع البيانات ، وتستطيع التطبيقات أن تشكل القائمة من خلال قاعدة بيانات أو أن تحفظ القائمة داخل قاعدة بيانات ، وتستطيع أن تجري على العقد العمليات التي ترغبها ، تستطيع أن تضيف حساب شخص بأي مكان من القائمة ، تستطيع حذف حساب شخص متى ما أردت ومن أي مكان بالقائمة ، تستطيع ترتيب الحسابات ، عرضها ، تحريرها كذلك تستطيع تمديد القائمة وإضافة عناصر جديدة من العمليات المختلفة بكل سهوله وكفاءة، على عكس المصفوفات التي تتطلب تحريك العناصر.

H.W:

1- write program to calculate the summation of data in linked list nodes.

2-write program to construct a linked list of student consist of **Name , Age, Class** felids.