جامعة المنارة
كلية الهندسة
قسم الهندسة المعلوماتية

مقرر الخوارزميات و بنى المعطيات1

جلسة العملي الخامسة

اللائحة المترابطة linked list

- تعريف صف لتمثيل عقد اللائحة :

```
class Node
{ public: Node(): next(NULL){};
  int data; Node *next;
};
```

- تعريف صف لتمثيل اللائحة المترابطة مع تعريف الباني :

```
class linklist{
private:Node *first;
public:
linklist():first(NULL){}
```

تعريف باقي العمليات الأساسية للتعامل مع اللائحة المترابطة ضمن الصف :

- التنقل ضمن اللائحة و طباعة عناصرها :

```
void display() {
  Node *c=first;
  while(c!=NULL){
       cout<<c->data<<"  ";
       c=c->next;}
  cout<<endl; }
```

- **إضافة عنصر إلى بداية اللائحة :**

```
void addfirst(int n) {
   Node *p=new Node ;
   p->data=n;
   p->next=first;
   first=p;;
}
```

- **إضافة عنصر إلى نهاية اللائحة :**

```
void addlast(int n) {
    if(first==NULL)
        addfirst(n);
  else{
  Node* c=first;
  while(c->next!=NULL)
    c=c->next;
  Node * p=new Node ;
  p->data=n;
  c->next=p;
  }
}
```

- **تابع لإدخال مجموعة من الأعداد الصحيحة من لوحة المفاتيح و إضافتها إلى اللائحة المترابطة من الخلف بحيث ينتهي الإدخال عند إدخال العدد 0 .**

```
void additem() {
  Node *c;
  while(1){
  int n;  cin>>n;
  if(n==0)break;
  Node *p=new Node ;
  p->data=n;
  if(first==NULL)
  {
  p->next=first;
  first=p;
  c=first;
  }
  else
    {
       c->next=p;
       c=c->next;
    }
  }}
```

- تابع لترتيب اللائحة :

```
void sort( ) {Node *current=first,*tempptr;
  while(current!=NULL) {tempptr=current->next;
        while(tempptr!=NULL) {if(current->data>tempptr->data)
         {int temp=current->data;
          current->data=tempptr->data;
          tempptr->data=temp;}
          tempptr=tempptr->next; }
  current=current->next;}
 }
```

- تابع لحشر عنصر في لائحة مرتبة :

```
void insert(int m) {
   Node *p=new Node ;
   p->data=m;
   Node *befor,*after=first;
  if(p->data < first->data)
   {p->next=first;
    first=p; } // ins  first
  else
   {
     while(after!=NULL)
```

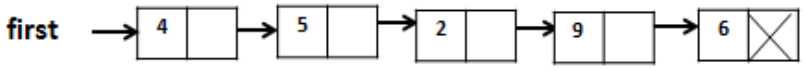
```
        {
          if(p->data < after->data)break;  // find the position
          befor =after;
          after=after->next;  // move to next node
        }
      p->next=after;
      befor ->next=p;
    }
  }
```

- **البرنامج الرئيسي:**

```
int main() {
    linklist li; int t;
    li.additem();
    cout<<"items are : ";
    li.display();
    li.sort();
    cout<<"items after sorting : ";
    li.display();
    cout<<"enter number :";
    cin>>t;
    li.insert(t);              // insert node
```

```
cout<<"items are : ";
li.display();        // print data
}
```

تمرين: أوجد خرج المقطع البرمجي المطبق على اللائحة التالية




```
int s = 0;
LinkList * p= first -> next;
while(p -> next != NULL)
{
s = s + p -> data;
p = p -> next;
}
cout<< "s=" << s;
```

الحل :

S = 16