

```

#include <iostream>
using namespace std;
class lista;
class nodo
{
friend class lista;
private:
int info;
nodo *sig;
public:
nodo();
nodo(int,nodo *);
};
nodo::nodo():info(0),sig(0){ }
nodo::nodo(int x,nodo *p=0):info(x),sig(p){ }
class lista
{
private:
nodo *p;
void AltaRe( nodo * &, int);
void BajaRe (nodo * &,int);
bool EstaRe (const nodo *, int)const;
void EmiteRe (const nodo *)const ;
public:
lista(); //constructor sin argumentos; agregar el constructor copia
~lista(); //destructor
void altafin(int); // alta al final
void altaprin(int); //alta al principio
void baja(int); //baja de un elemento determinado
bool esta(int) const ; // retorna un bool indicando si el entero pasado por argumento está en la lista
void emite() const; //emite la lista
};

lista::lista():p(0){ }
void lista::altafin(int x)
{
AltaRe(p, x);
}

void lista::altaprin(int x)
{
p=new nodo(x,p);
}

void lista::baja(int x)
{
BajaRe(p,x);
}
lista::~~lista()
{
if (p)
{
nodo *aux=p;
while (aux)
{
p=p->sig;

```

```

delete aux;
aux=p;
}
}
}
bool lista::esta(int x)const
{
return (EstaRe(p,x));
}

void lista::AltaRe( nodo * &q, int x)
{
if (!q) q=new nodo(x,0);
else AltaRe(q->sig,x);
}
void lista::BajaRe (nodo * &q ,int x)
{
if(q)
    if (q->info ==x) {
        nodo * aux=q;
        q=q->sig;
        delete q;
    }
    else BajaRe(q->sig , x);
}
bool lista::EstaRe (const nodo * q, int x) const
{
if (q)
    {if(q->info ==x) return true;
    else return (EstaRe(q->sig, x));
    }
return false;
}

void lista::emite() const
{
EmiteRe(p);
}
void lista::EmiteRe(const nodo * q) const
{
if(q)
    {
    cout<<q->info<<" ";
    EmiteRe(q->sig);
    }
}
void main()
{
lista l;
cout<<"\n alta al final de 1\n";
l.altafin(1);
l.emite();
cout<<"\n alta al final de 2\n";
l.altafin(2);
l.emite();
cout<<"\n alta al final de 3\n";

```

```
l.altafin(3);
l.emite();
cout<<"\n alta al final de 4\n";
l.altafin(4);
l.emite();
cout<<"\n alta al principio de 5\n";
l.altaprin(5);
l.emite();
cout<<"\n baja 4\n";
l.baja(4);
l.emite();
cout<<"\n baja 1\n";
l.baja(1);
l.emite();
int i;
cin>>i;
}
```