

# **Sessió 10**

Tema 2

## Exercici 2.7:

2.7 Donades les següents declaracions de les classes Node i Llista:

```
class Node
{ public:
    Node();
    ~Node();
    Node (const int &valor)
    Node *getNext()
    void setNext(Node *next)
    int getValor()
    void setValor(const int &valor)
private:
    int m_valor;
    Node *m_next;
};
```

```
class Llista
{public:
    Llista();
    ~Llista();
    bool esBuida() const;
    Node *getInici() const;
    Node *insereixNext(
        const int& valor,
        Node *posicio);
    Node *eliminaNext(Node*
posicio);
private:
    Node* m_primer;
};
```

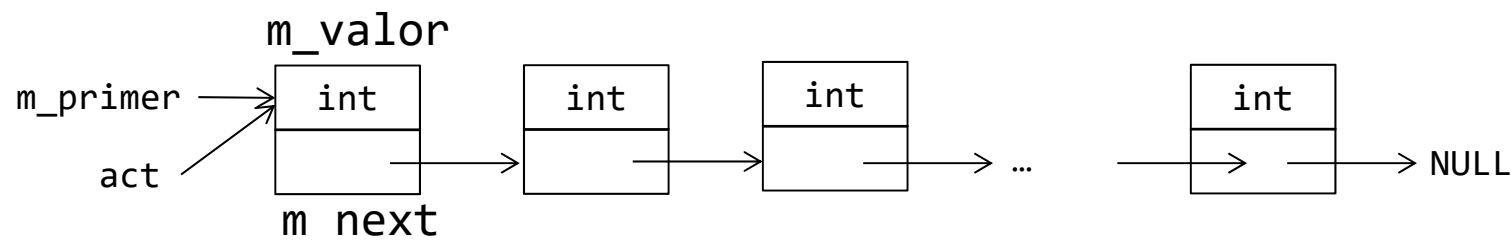
## Exercici 2.7.2: implementa mètode suma.

2.7.2 Implementa un mètode que retorni la suma de tots els elements de la llista. La seva declaració és la següent:

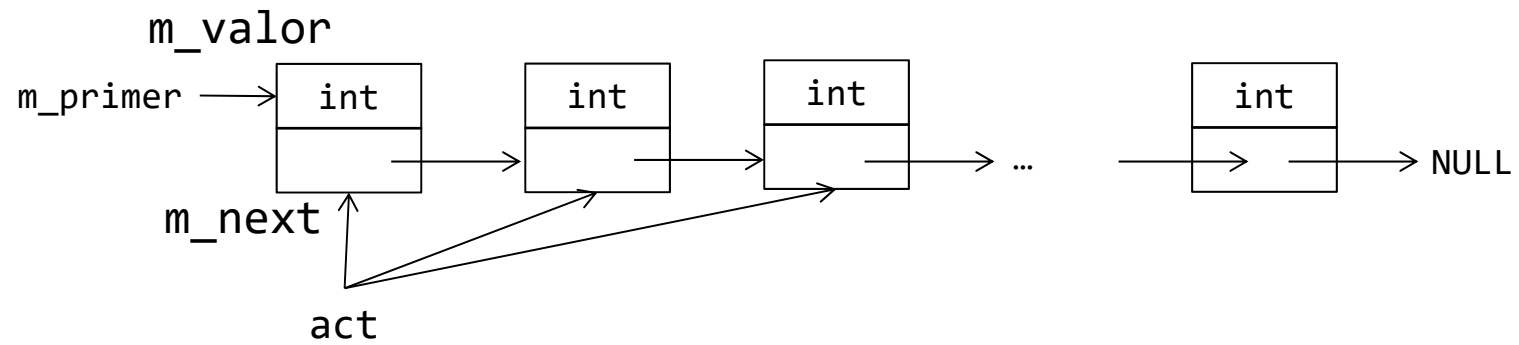
- `int sumaElements() const;`

```
class Node
{ public:
    Node();
    ~Node();
    Node (const int &valor)
    Node *getNext()
    void setNext(Node *next)
    int getValor()
    void setValor(const int &valor)
private:
    int m_valor;
    Node *m_next;
};
```

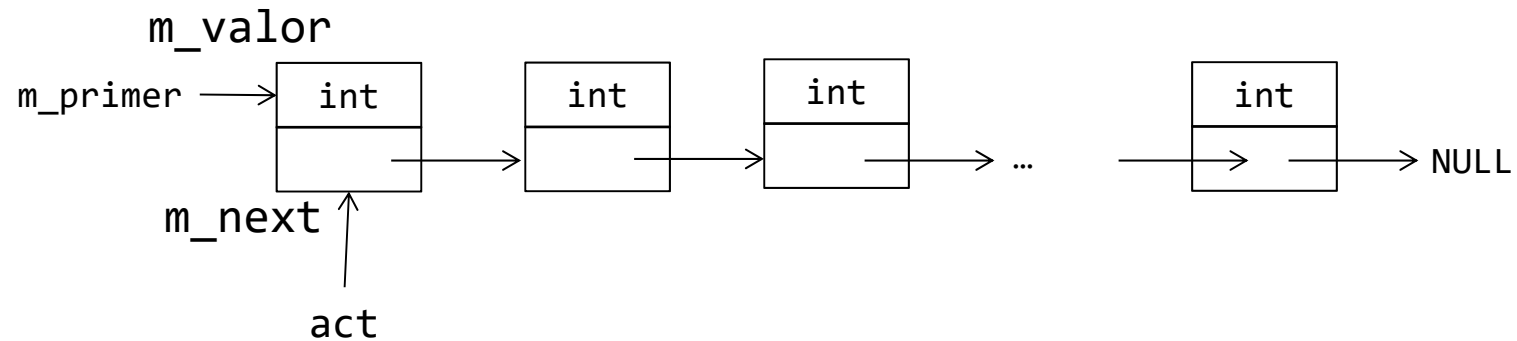
```
class Llista
{public:
    Llista();
    ~Llista();
    bool esBuida() const;
    Node *getInici() const;
    Node *insereixNext(
        const int& valor,
        Node *posicio);
    Node *eliminaNext(Node* posicio);
private:
    Node* m_primer;
```



## Exercici 2.7.2: Solució



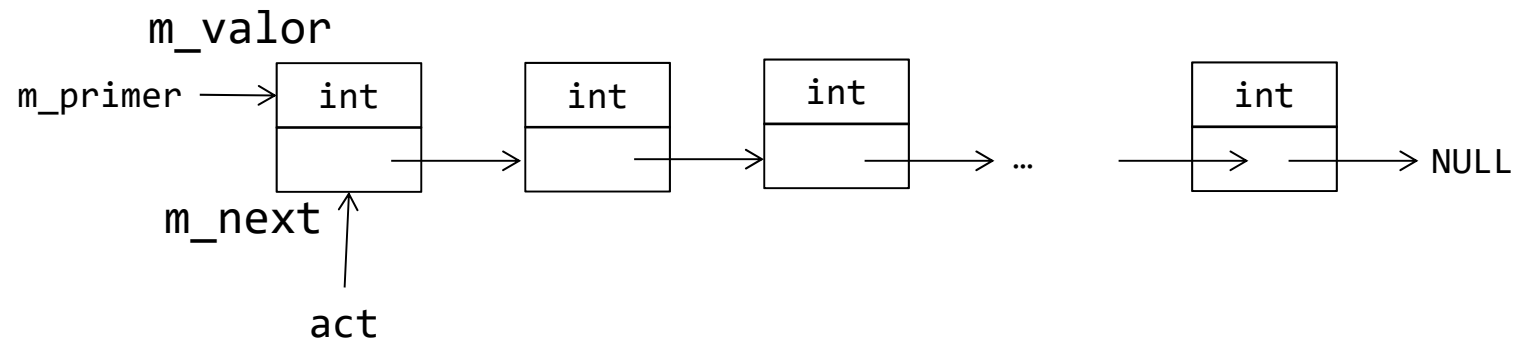
## Exercici 2.7.2: Solució



```
int Llista::sumaElements() const
{
    int suma = 0;
    Node *act = m_primer;
    while (act != NULL)
    {

    }
    return suma;
}
```

## Exercici 2.7.2: Solució

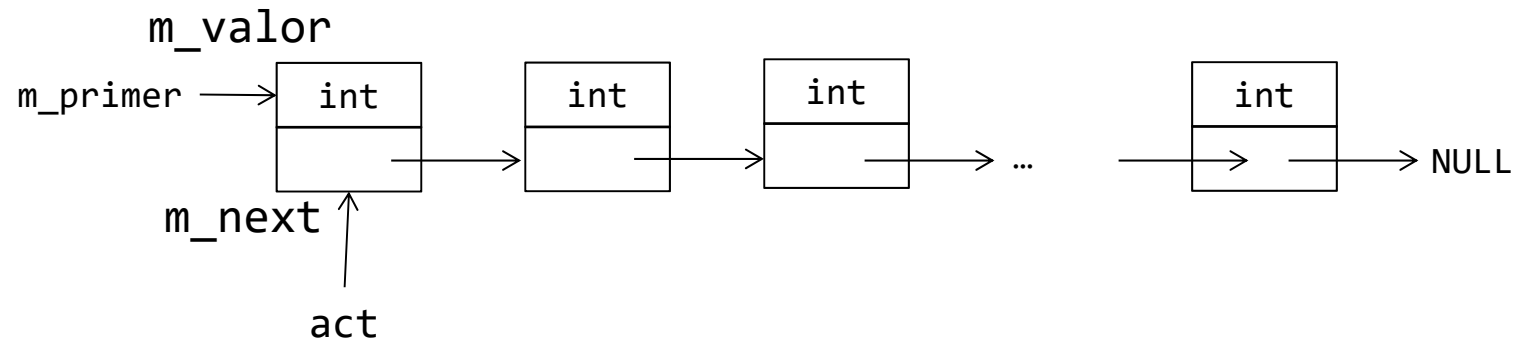


```
int Llista::sumaElements() const
{
    int suma = 0;
    Node *act = m_primer;
    while (act != NULL)
    {
        }

    return suma;
}
```

```
act->getValor();
act->getNext();
```

## Exercici 2.7.2: Solució



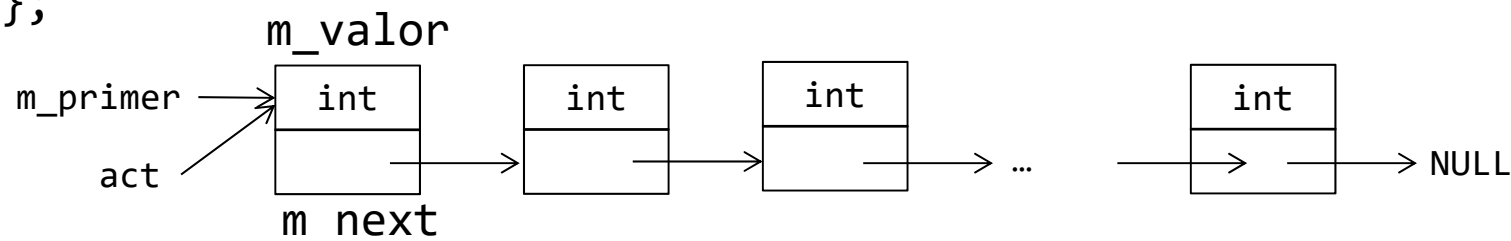
```
int Llista::sumaElements() const
{
    int suma = 0;
    Node *act = m_primer;
    while (act != NULL)
    {
        suma += act->getValor();
        act = act->getNext();
    }
    return suma;
}
```

## Exercici 2.7.3: implementa mètode eliminaValor.

2.7.3 Implementa un mètode per eliminar tots elements que siguin iguals al valor que es passa com a paràmetre. La seva declaració és la següent:

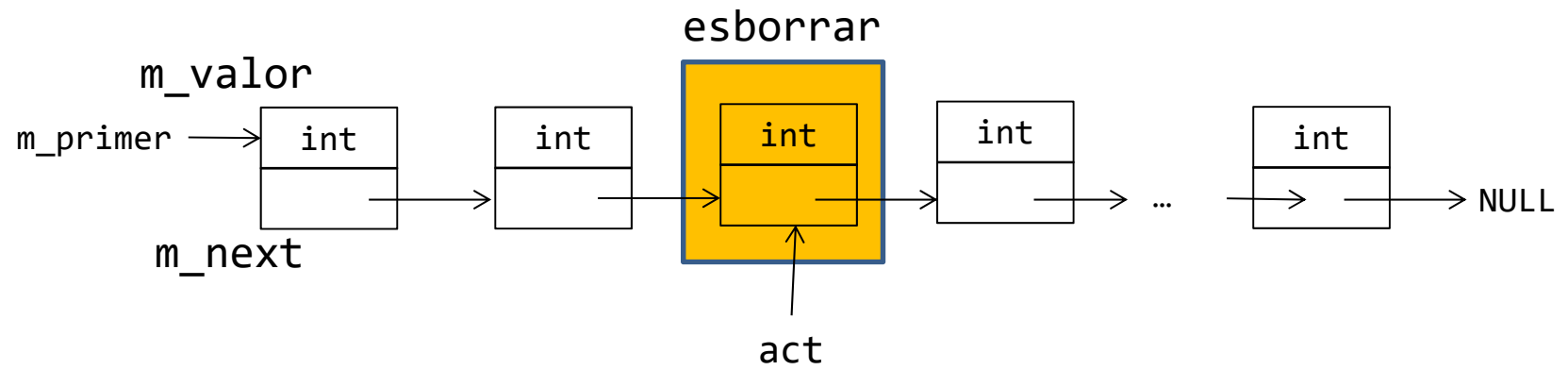
```
• void eliminaValor(int valor);  
class Node  
{ public:  
    Node();  
    ~Node();  
    Node (const int &valor)  
    Node *getNext()  
    void setNext(Node *next)  
    int getValor()  
    void setValor(const int &valor)  
private:  
    int m_valor;  
    Node *m_next;  
};
```

```
class Llista  
{public:  
    Llista();  
    ~Llista();  
    bool esBuida() const;  
    Node *getInici() const;  
    Node *insereixNext(  
        const int& valor,  
        Node *posicio);  
    Node *eliminaNext(Node*  
posicio);  
private:  
    Node* m_primer;  
};
```

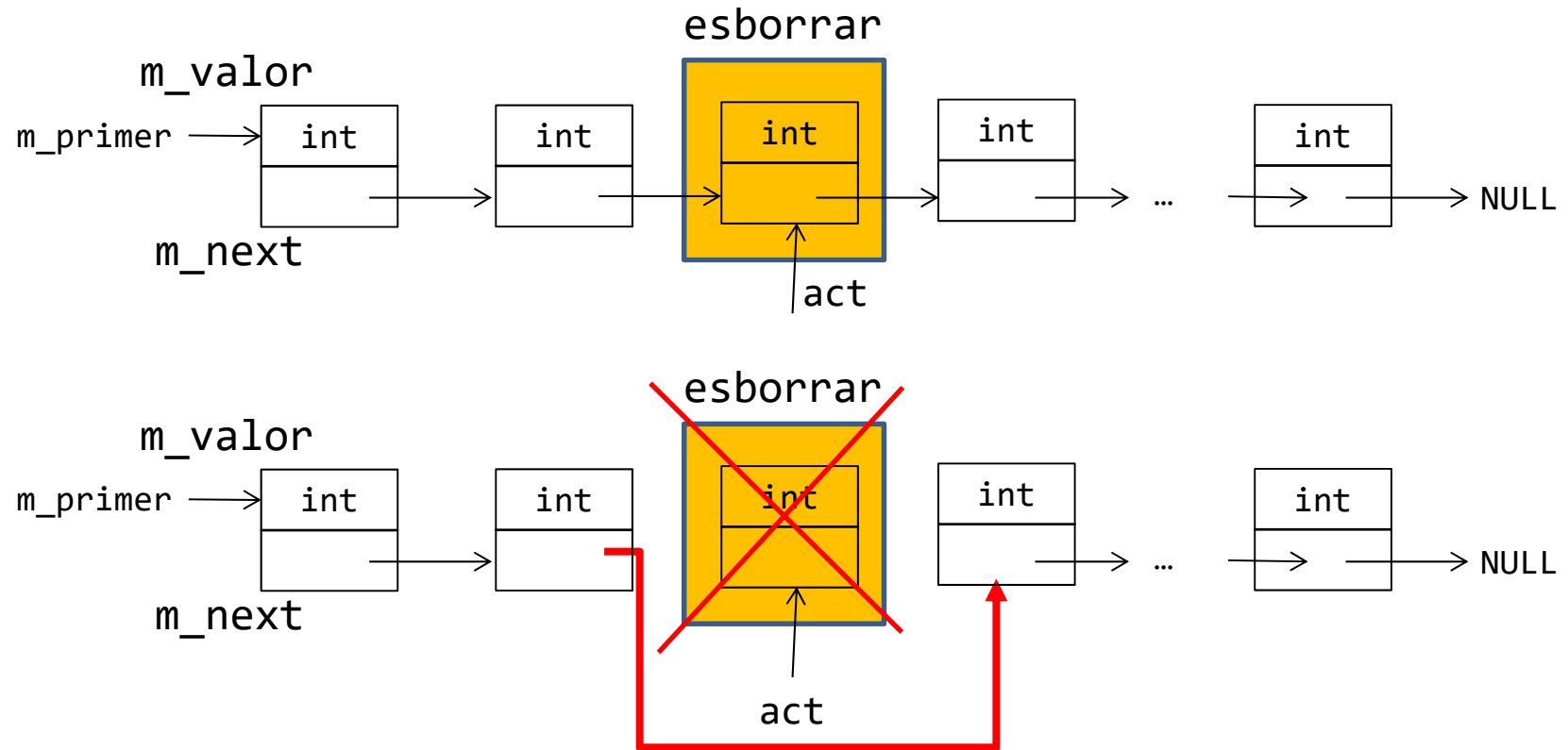




## Exercici 2.7.3: Solució

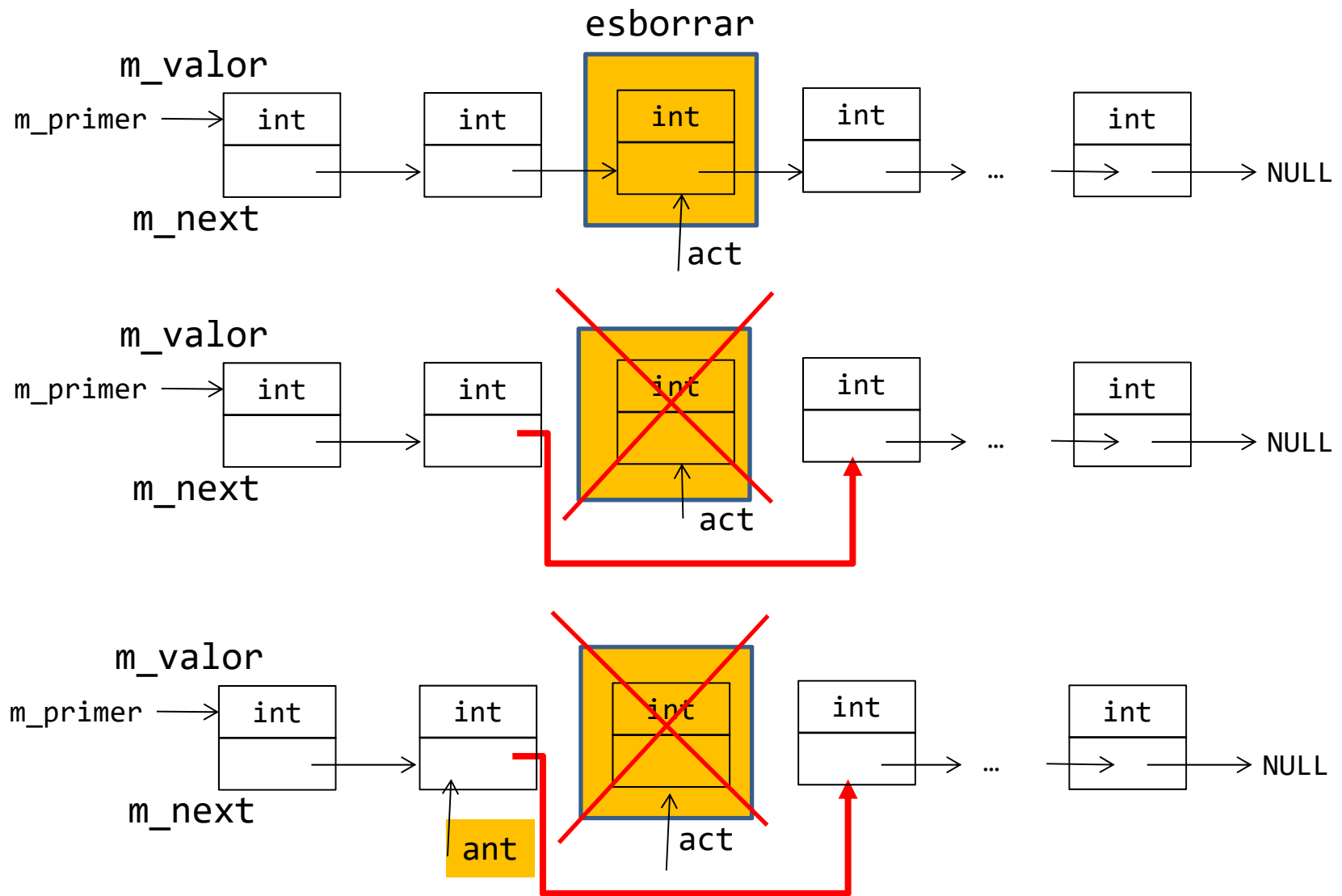


## Exercici 2.7.3: Solució



Com puc fer aquest canvi a partir de `act`?  
No es pot

## Exercici 2.7.3: Solució



Solució a més de l'apuntador `act` (actual) mantenim el `ant` (anterior)

## Exercici 2.7.3: Solució

```
void Llista::eliminaValor(int valor)
{
    Node *act = m_primer;
    Node *ant = NULL;
    while (act != NULL)
    {
        if (act->getValor() == valor)
        {
            if (ant != NULL)
            {
                ant->setNext(act->getNext());
                delete act;
                act = ant->getNext();
            }
            else
            {
                m_primer = act->getNext();
                delete act;
                act = m_primer;
            }
        }
        else
        {
            ant = act;
            act = act->getNext();
        }
    }
}
```

Mètode eliminaNext  
de Llista ja fet

## Exercici 2.7.3: Solució

```
void Llista::eliminaValor(int valor)
{
    Node *act = m_primer;
    Node *ant = NULL;
    while (act != NULL)
    {
        if (act->getValor() == valor)
        {
            act=eliminaNext(ant);
        }
        else
        {
            ant = act;
            act = act->getNext();
        }
    }
}
```