Sessió 10

Tema 2

Exercici 2.7:

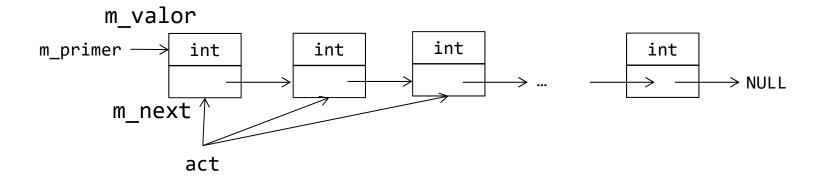
2.7 Donades les següents declaracions de les classes Node i Llista:

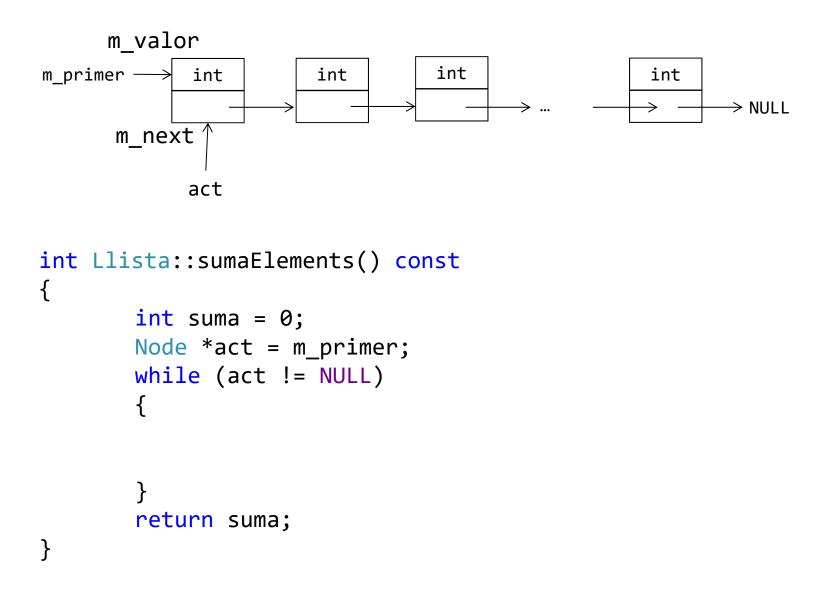
```
class Llista
class Node
                                      {public:
{ public:
                                        Llista();
  Node();
                                        ~Llista();
   ~Node();
                                        bool esBuida() const;
   Node (const int &valor)
                                        Node *getInici() const;
  Node *getNext()
                                        Node *insereixNext(
   void setNext(Node *next)
                                               const int& valor,
   int getValor()
   void setValor(const int &valor)
                                               Node *posicio);
                                        Node *eliminaNext(Node*
 private:
                                      posicio);
   int m valor;
                                      private:
   Node *m next;
                                        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
                                       class Llista
{ public:
                                       {public:
   Node();
                                         Llista();
   ~Node();
                                         ~Llista();
   Node (const int &valor)
                                         bool esBuida() const;
   Node *getNext()
                                         Node *getInici() const;
   void setNext(Node *next)
                                         Node *insereixNext(
   int getValor()
                                               const int& valor,
                                               Node *posicio);
   void setValor(const int &valor)
 private:
                                         Node *eliminaNext(Node* posicio);
   int m valor;
                                       private:
   Node *m next;
                                         Node* m primer;
};
                                       };
             m valor
  m primer
               int
                           int
                                      int
                                                         int
                                                                → NULL
     act
              m next
```





```
m_valor
m_primer —
                                  int
             int
                        int
                                                     int
                                                           \rightarrow NULL
      m_next 
            act
int Llista::sumaElements() const
        int suma = 0;
        Node *act = m_primer;
        while (act != NULL)
                                         act->getValor();
                                         act->getNext();
        return suma;
```

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

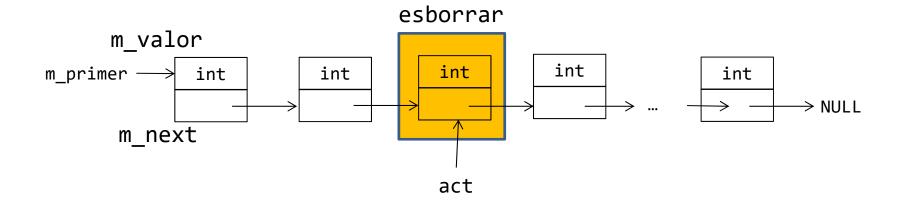
Exercici 2.7.3: implementa mètode elimina Valor.

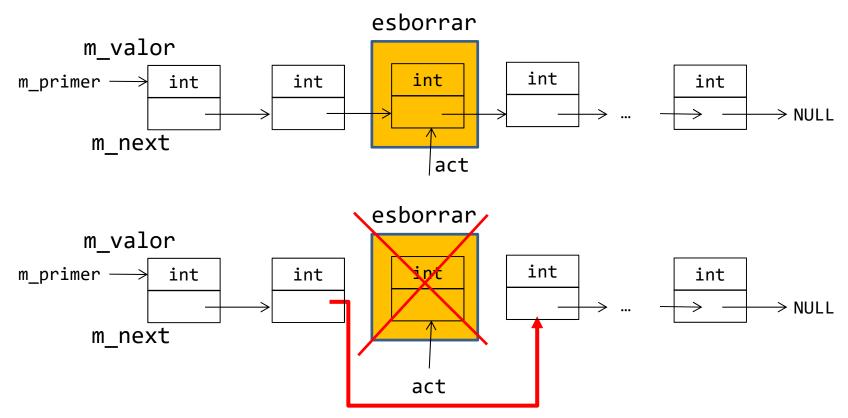
2.7.3 Implementa un mètode per eliminar tots elements que siguin iguals al valor que es passa com a paràmetre. La seva declaració és la següent:

```
class Llista

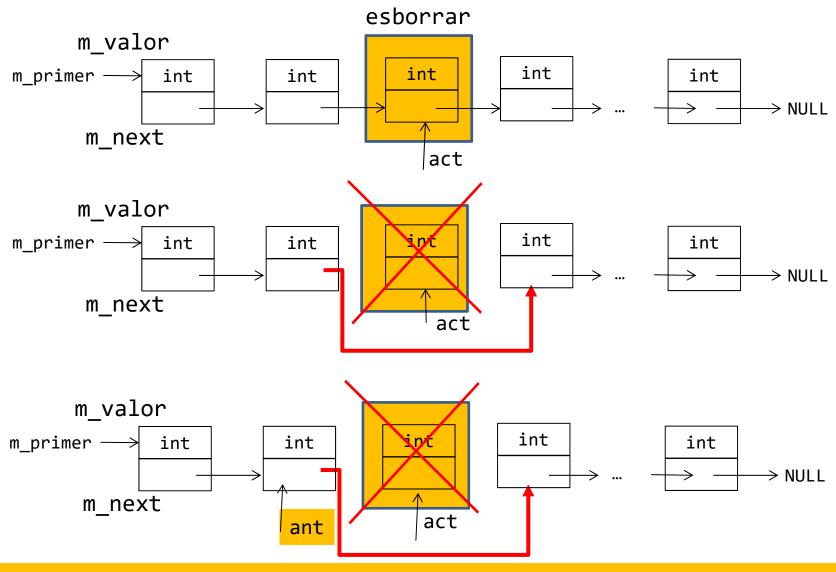
    void eliminaValor(int valor);

                                        {public:
class Node
                                          Llista();
{ public:
                                          ~Llista();
   Node();
                                          bool esBuida() const;
   ~Node();
                                          Node *getInici() const;
   Node (const int &valor)
                                          Node *insereixNext(
   Node *getNext()
                                                const int& valor,
   void setNext(Node *next)
                                                Node *posicio);
   int getValor()
                                          Node *eliminaNext(Node*
   void setValor(const int &valor)
                                        posicio);
 private:
                                        private:
   int m valor;
                                          Node* m primer;
   Node *m next;
                                        };
};
          m_valor
m primer
             int
                       int
                                 int
                                                  int
                                                         → NULL
    act
           m next
```





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



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

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

```
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();
```